PF 980017

CITED BY APPLICANT

(12) UK Patent Application (19) GB (11) 2 264 842 (13) A

(43) Date of A publication 08.09.1993

(21) Application No 9113071.6

(22) Date of filing 18.06.1991

(30) Priority data
(31) 9026320 (32) 04.12.1990 (33) GB

(51) INT CL[5]
H04K 1/04, H04L 7/00

(52) UK CL (Edition L)
H4L LBSF L1H9
H4P PPF

(56) Documents cited
None

(58) Field of search
UK CL (Edition K) H4L LBSF
INT CL[5] H04B, H04K, H04L
Online databases: WPI

(71) Applicant
Roke Manor Research Limited

(Incorporated in the United Kingdom)

Roke Manor, Romsey, Hampshire, SO51 0ZN,
United Kingdom

(72) Inventor
Anthony Peter Hulbert

(74) Agent and/or Address for Service
Norman Ernest Fish
Siemens Group Services Limited,
Intellectual Property Department, Roke Manor,
Old Salisbury Lane, Romsey, Hampshire, SO51 0ZN,
United Kingdom

(54) Synchronization of nodeless frequency-hopping communications system

(57) A frequency hopping distributed (masterless) radio communication system comprises a plurality of transceivers each having a clock error prior to synchronisation not exceeding ±N seconds with respect to absolute time wherein the following operation conditions are arranged to obtain.

a) synchronisation transmissions are made by each transceiver at random time during an interval NM seconds (e.g. 3N) defined at the centre of contiguous windows of (4+M)N seconds (e.g. 0–6N, where M=2), one transmission per window, the synchronisation transmission and reception frequency being the same f_0, f_1 for all transceivers during each window and varied pseudo-randomly from window to window. Each synchronisation transmission is arranged to include data indicative of the time of transmission within the interval NM seconds the origin of transmission;

b) all transceivers are arranged to respond to the reception of a synchronisation transmission with an acknowledgement (unless already in synchronisation with the transceiver which made the synchronisation transmission) upon receipt of which the transceiver which originated the synchronisation transmission is arranged to compute, in dependence upon the transmission and reception times of the acknowledgement transmission a common time which all radio transceivers involved are to assume; and,

c) the synchronisation transmission source transceiver is arranged to direct update command transmissions to each of the acknowledging transceivers containing instructions for timing data updates to bring the transceivers into synchronisation with the common timing so that all transceivers achieve synchronisation.

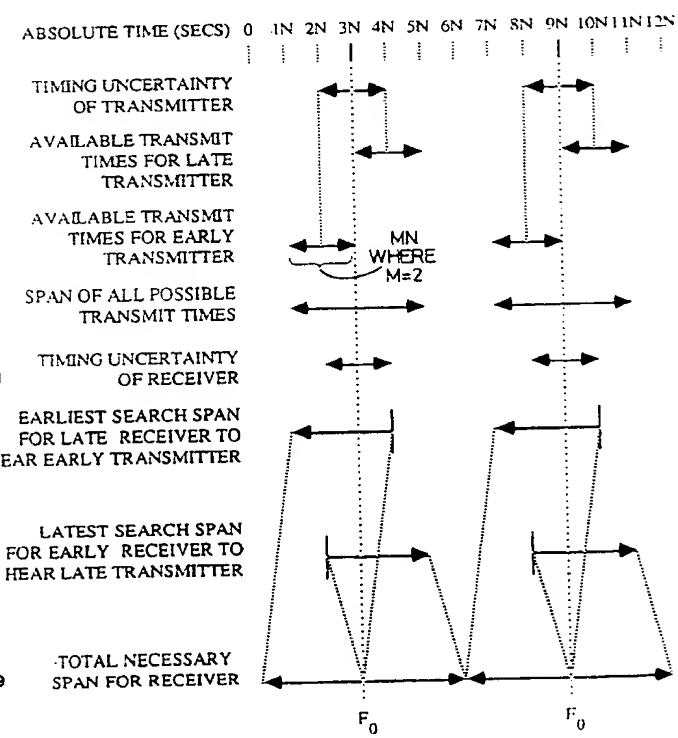
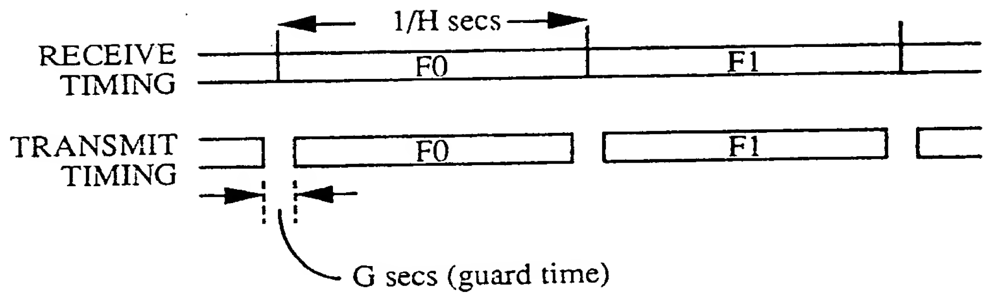


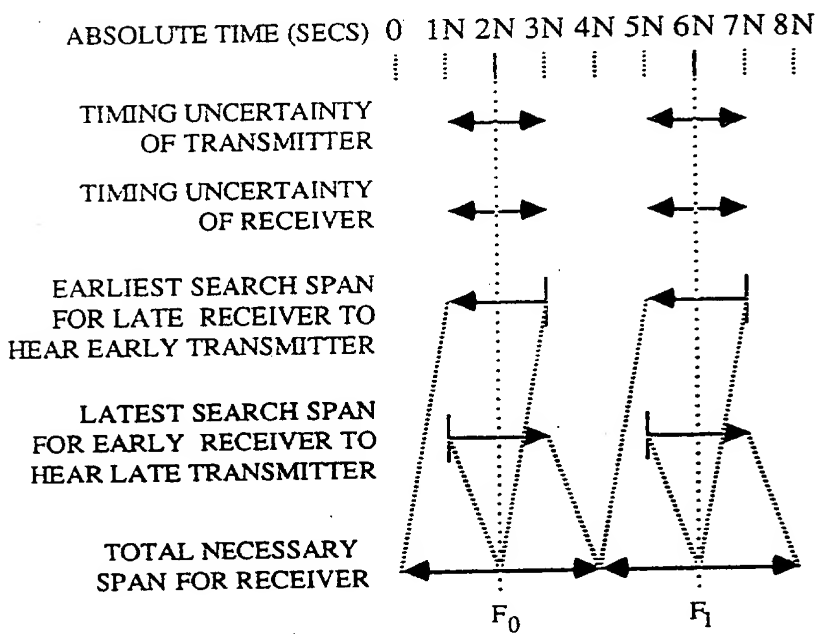
Fig. 3.

GB 2 264 842 A



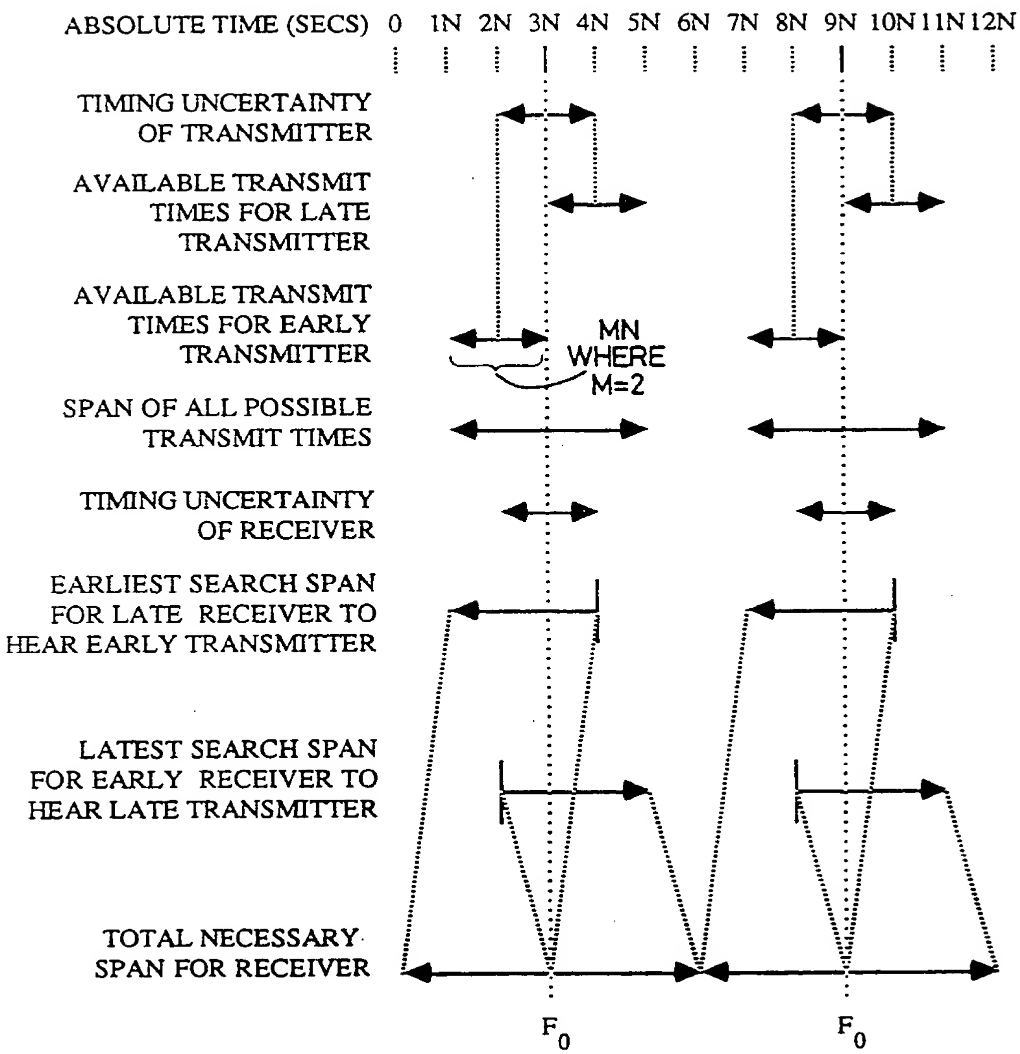
1/H secs
RECEIVE TIMING
F0
F1
TRANSMIT TIMING
F0
F1
G secs (guard time)

Fig.1.



ABSOLUTE TIME (SECS)
0
1N 2N 3N 4N 5N 6N 7N 8N
TIMING UNCERTAINTY OF TRANSMITTER
TIMING UNCERTAINTY OF RECEIVER
EARLIEST SEARCH SPAN FOR LATE RECEIVER TO HEAR EARLY TRANSMITTER
LATEST SEARCH SPAN FOR EARLY RECEIVER TO HEAR LATE TRANSMITTER
TOTAL NECESSARY SPAN FOR RECEIVER
F_0
F_1

Fig.2.



ABSOLUTE TIME (SECS)
0
1N
2N
3N
4N
5N
6N
7N
8N
9N
10N
11N
12N
TIMING UNCERTAINTY OF TRANSMITTER
AVAILABLE TRANSMIT TIMES FOR LATE TRANSMITTER
AVAILABLE TRANSMIT TIMES FOR EARLY TRANSMITTER
MN WHERE M=2
SPAN OF ALL POSSIBLE TRANSMIT TIMES
TIMING UNCERTAINTY OF RECEIVER
EARLIEST SEARCH SPAN FOR LATE RECEIVER TO HEAR EARLY TRANSMITTER
LATEST SEARCH SPAN FOR EARLY RECEIVER TO HEAR LATE TRANSMITTER
TOTAL NECESSARY SPAN FOR RECEIVER
F_0
F_0

Fig.3.

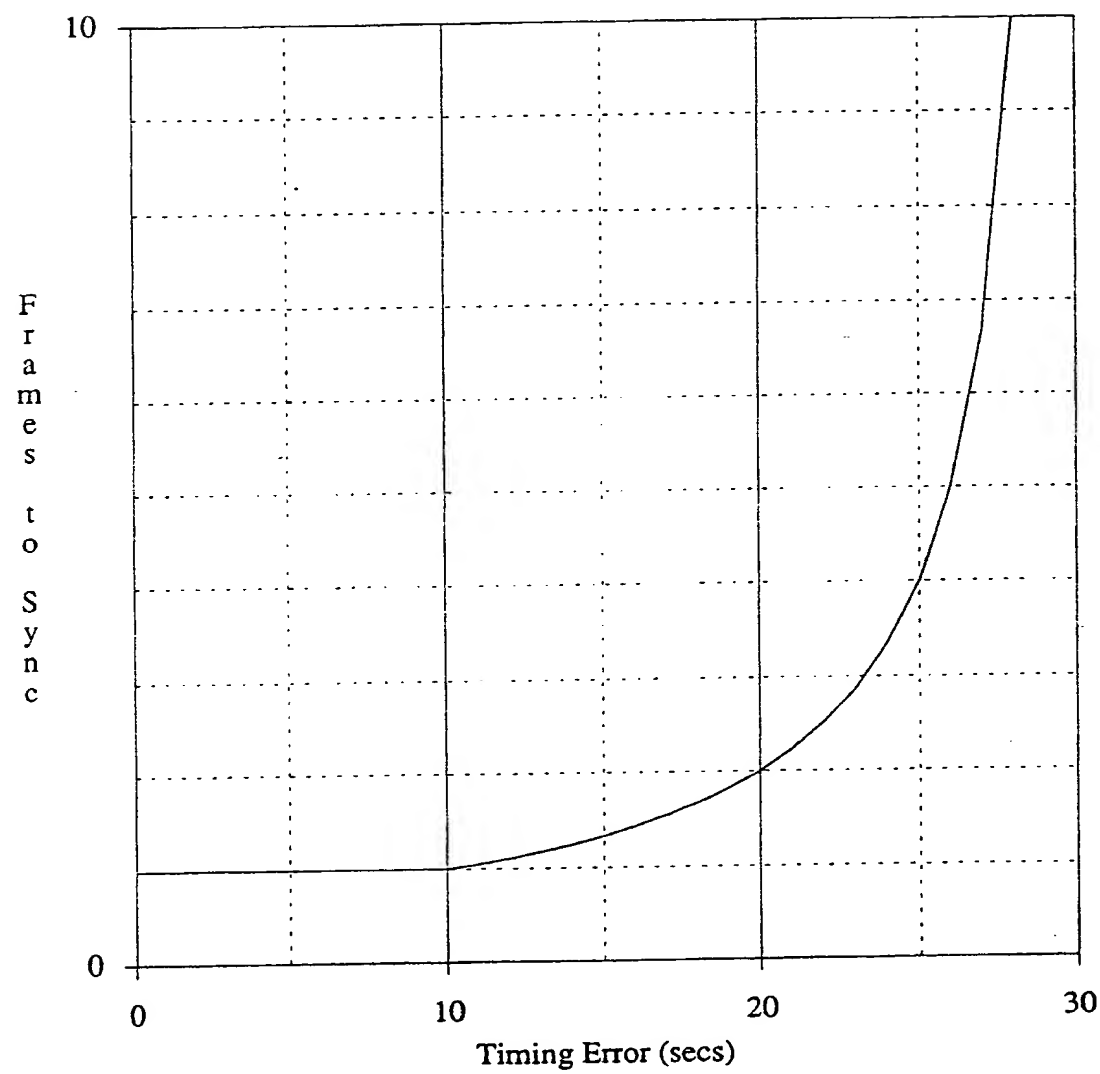


Fig. 4.

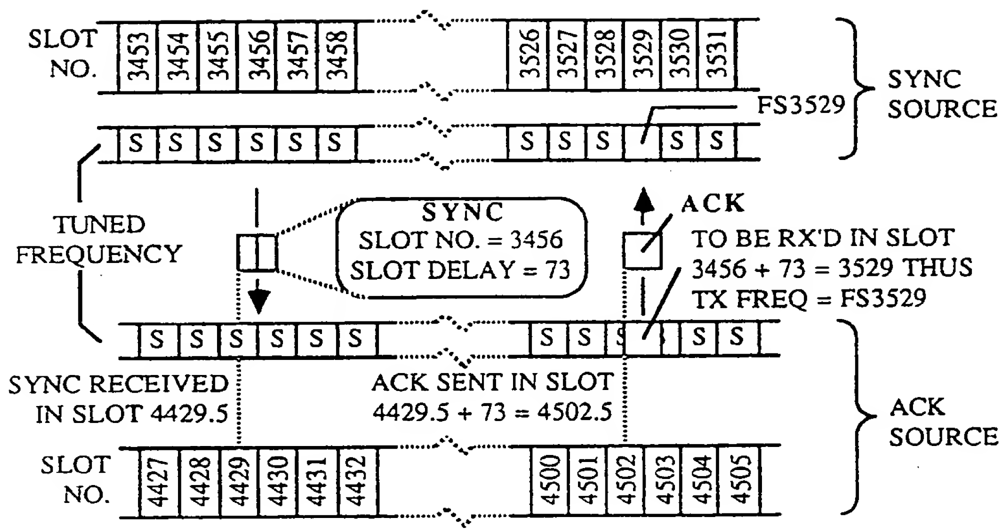

SLOT NO.
3453 3454 3455 3456 3457 3458
3526 3527 3528 3529 3530 3531
FS3529
SYNC SOURCE
S S S S S S S S S S S
TUNED FREQUENCY
SYNC
SLOT NO. = 3456
SLOT DELAY = 73
ACK
TO BE RX'D IN SLOT
3456 + 73 = 3529 THUS
TX FREQ = FS3529
S S S S S S S S S S S
SYNC RECEIVED IN SLOT 4429.5
ACK SENT IN SLOT
4429.5 + 73 = 4502.5
ACK SOURCE
SLOT NO.
4427 4428 4429 4430 4431 4432
4500 4501 4502 4503 4504 4505

Fig.5.

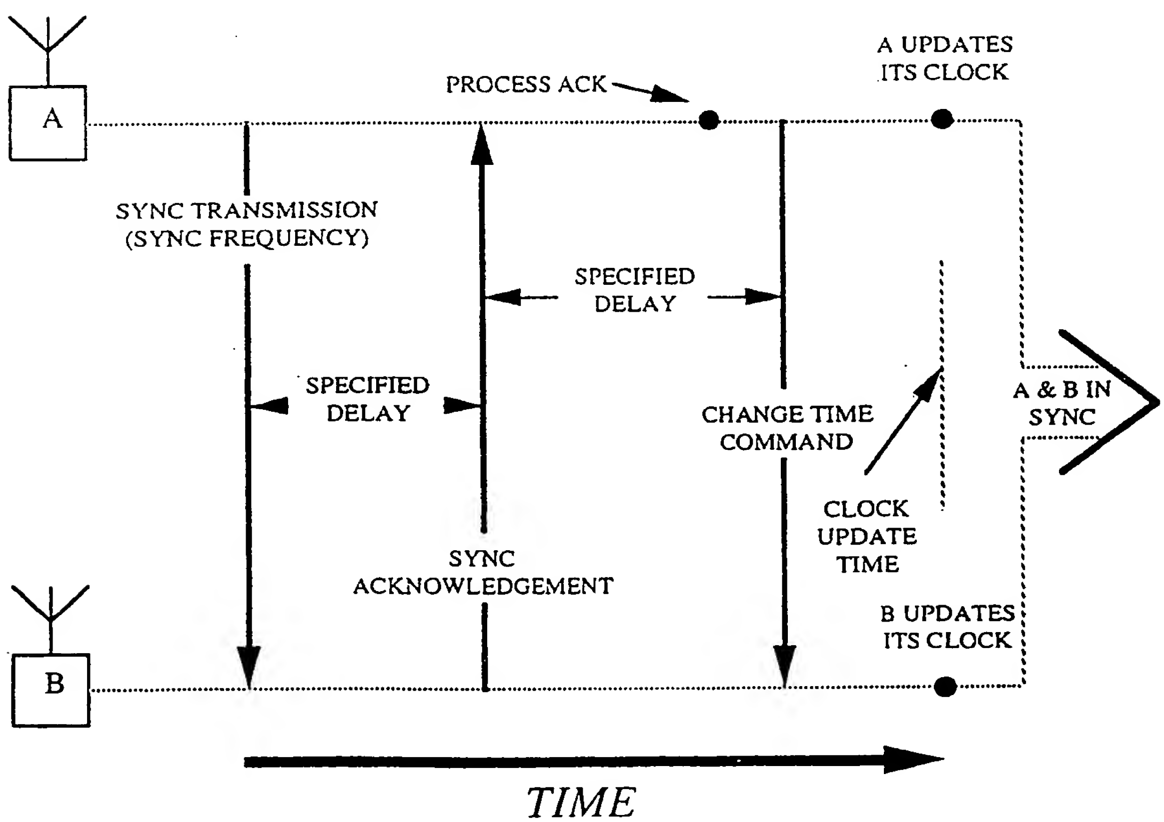

A
PROCESS ACK
A UPDATES ITS CLOCK
SYNC TRANSMISSION (SYNC FREQUENCY)
SPECIFIED DELAY
SPECIFIED DELAY
CHANGE TIME COMMAND
CLOCK UPDATE TIME
A & B IN SYNC
SYNC ACKNOWLEDGEMENT
B UPDATES ITS CLOCK
B
TIME

Fig.6.

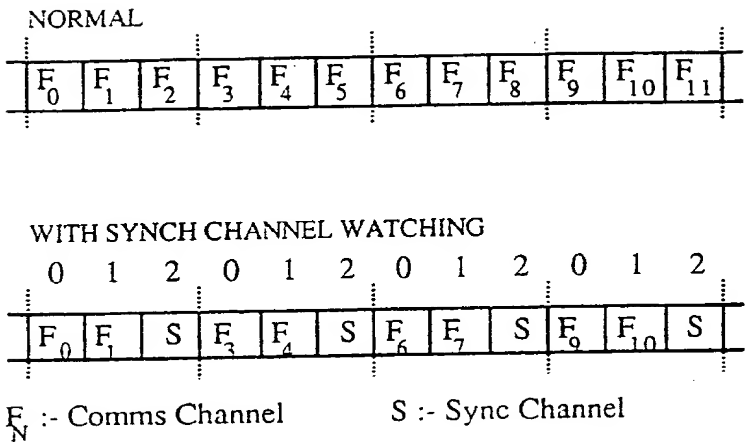


Fig.7.

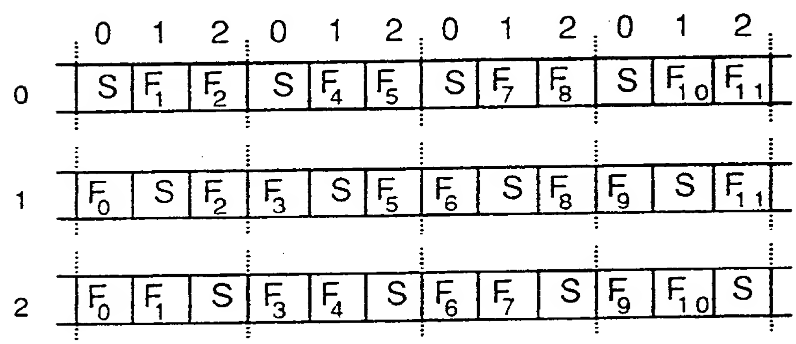


Fig.8.

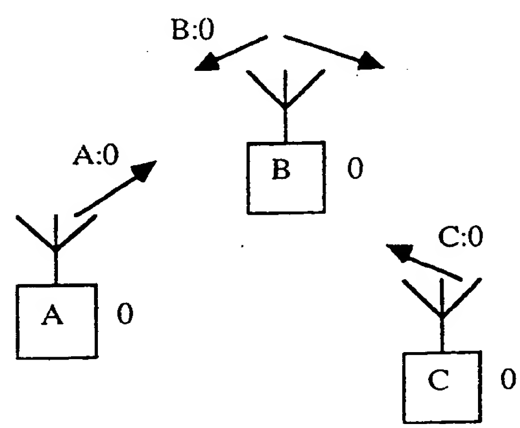


Fig.9.

Radio	Watch State
A	0
B	0
C	0

Fig.10.

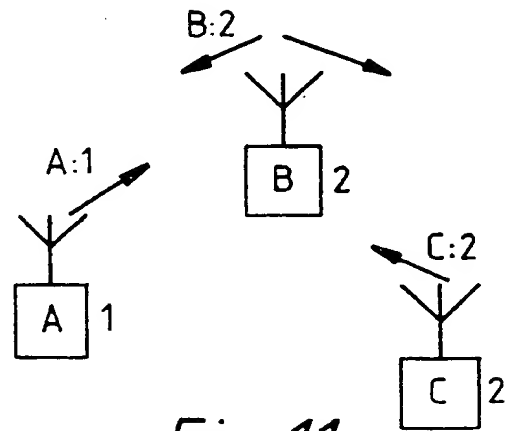


Fig.11.

Radio	Watch State
A	1
B	2
C	2

Fig.12.

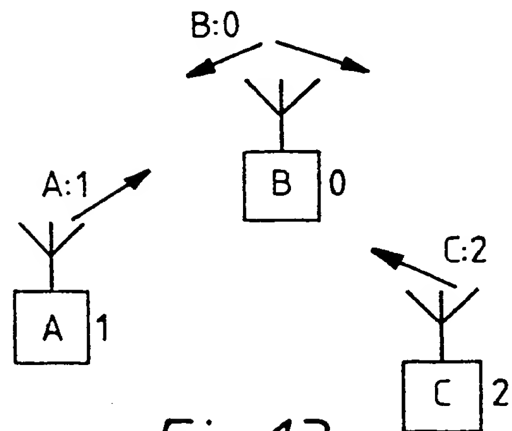


Fig.13.

Radio	Watch State
A	1
B	0
C	2

Fig.14.

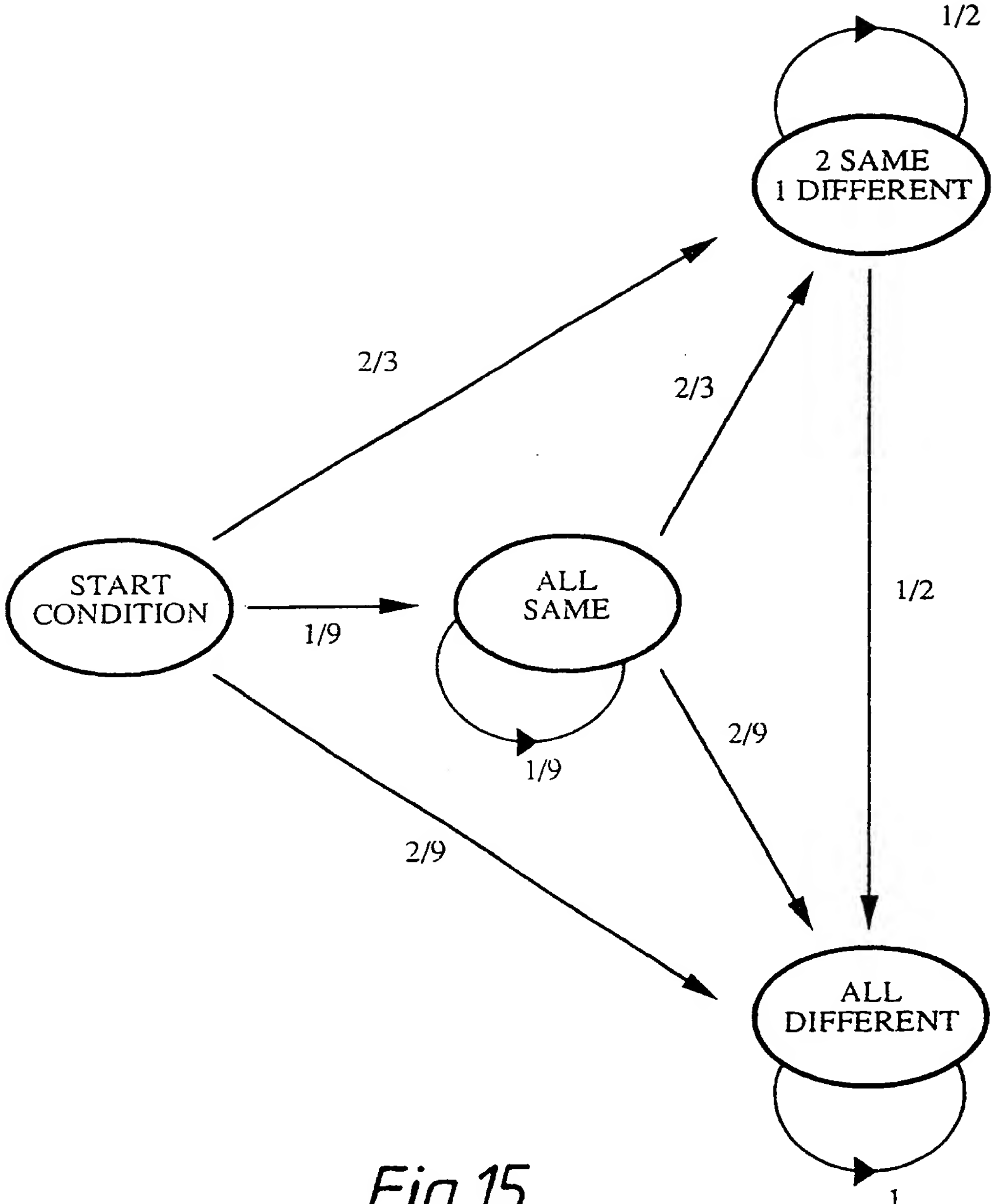

1/2
2 SAME
1 DIFFERENT
2/3
2/3
START
CONDITION
1/9
ALL
SAME
1/2
1/9
2/9
2/9
ALL
DIFFERENT
1

Fig.15.

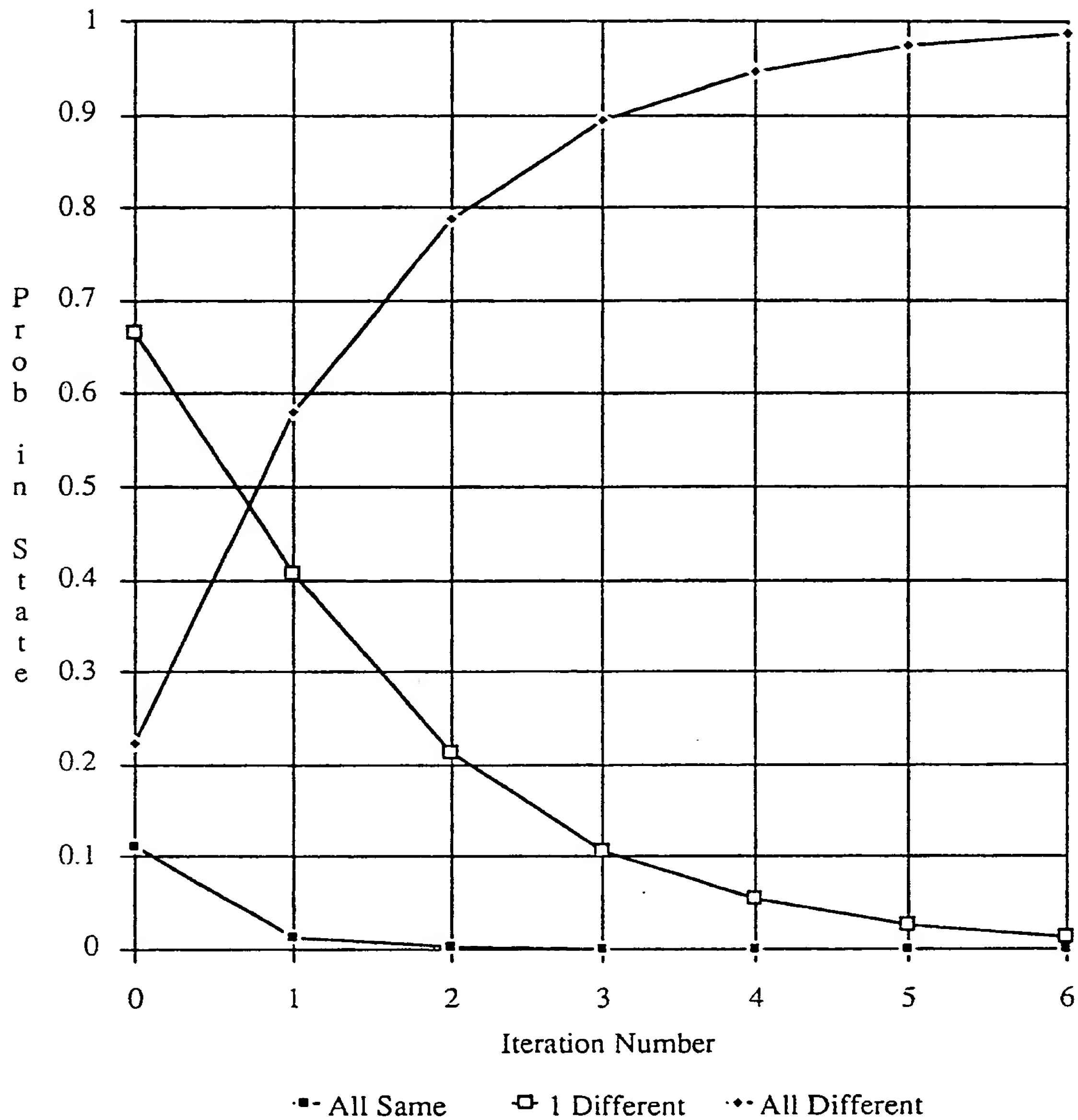

1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Prob in State
0
1
2
3
4
5
6
Iteration Number
All Same
1 Different
All Different

Fig.16.

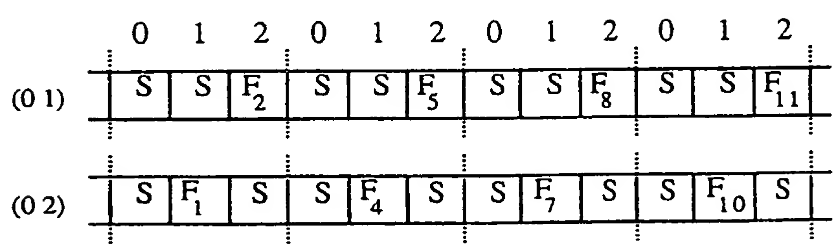

0 1 2 0 1 2 0 1 2 0 1 2
(0 1)
S S F2 S S F5 S S F8 S S F11
(0 2)
S F1 S S F4 S S F7 S S F10 S

Fig.17.

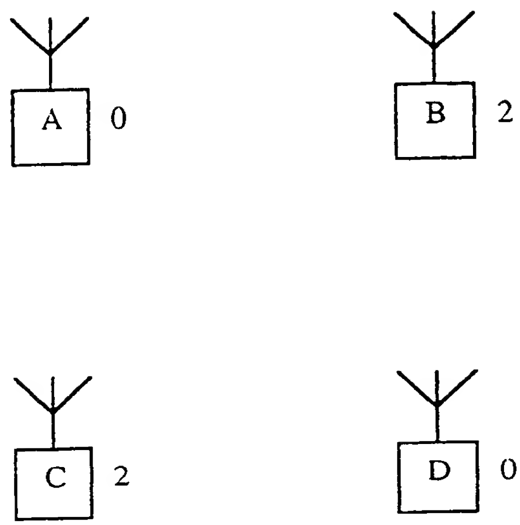


Fig.18.

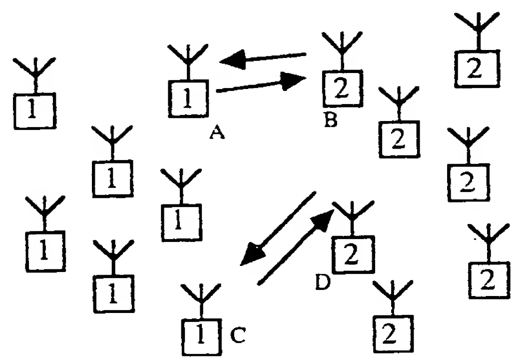


Fig.19.

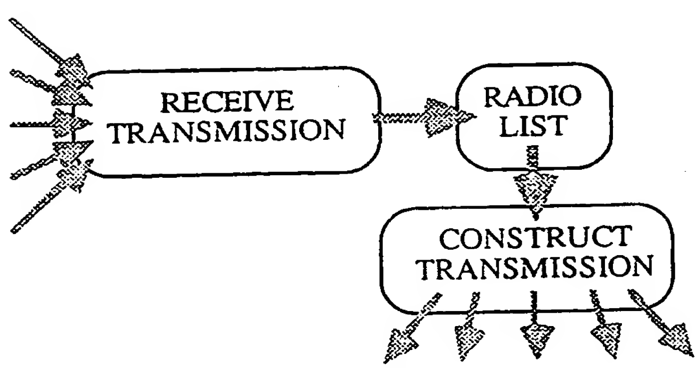


Fig. 20.

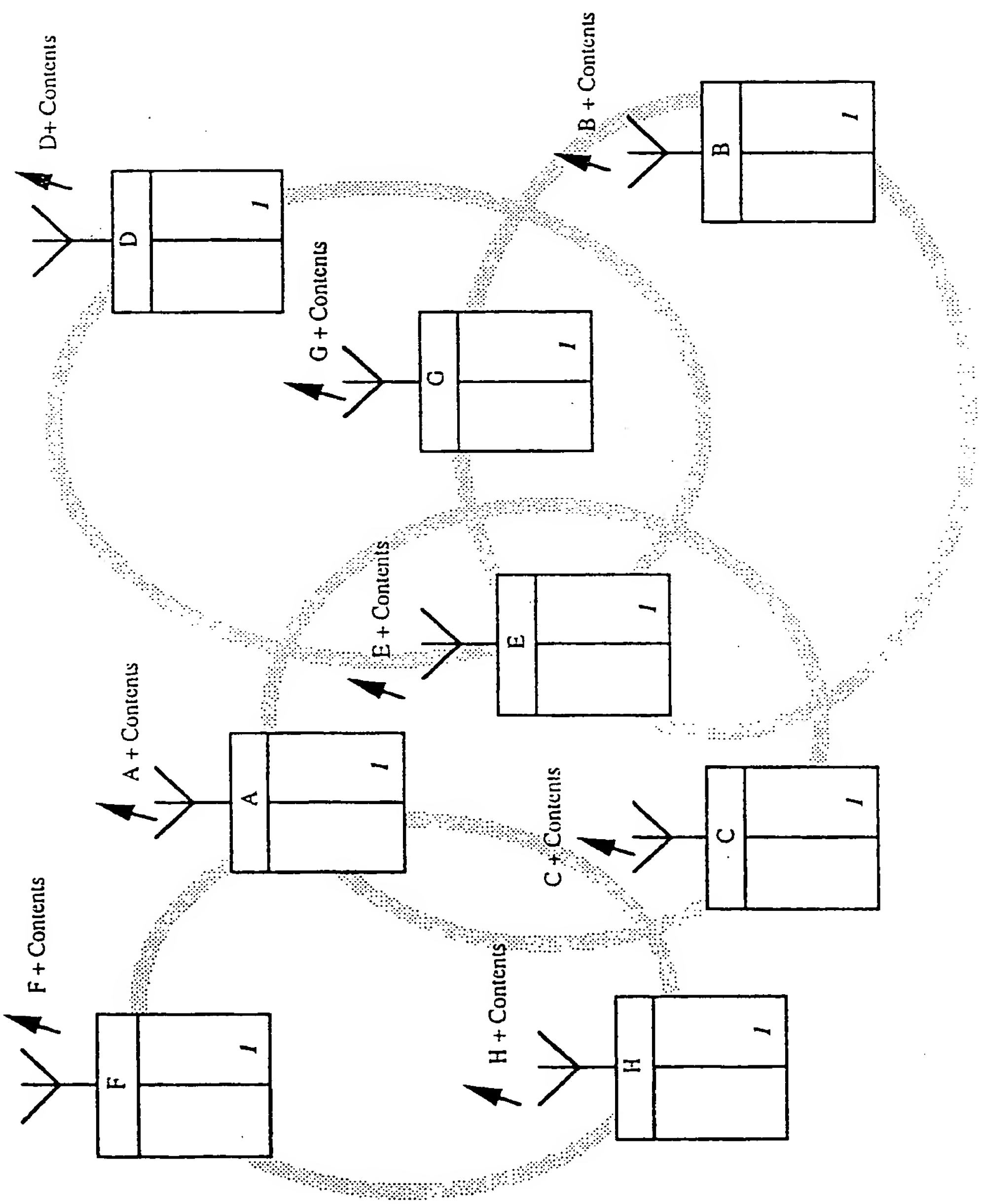

D+ Contents
D
B + Contents
B
G + Contents
G
E + Contents
E
A + Contents
A
C + Contents
C
F + Contents
F
H + Contents
H

Fig. 21.

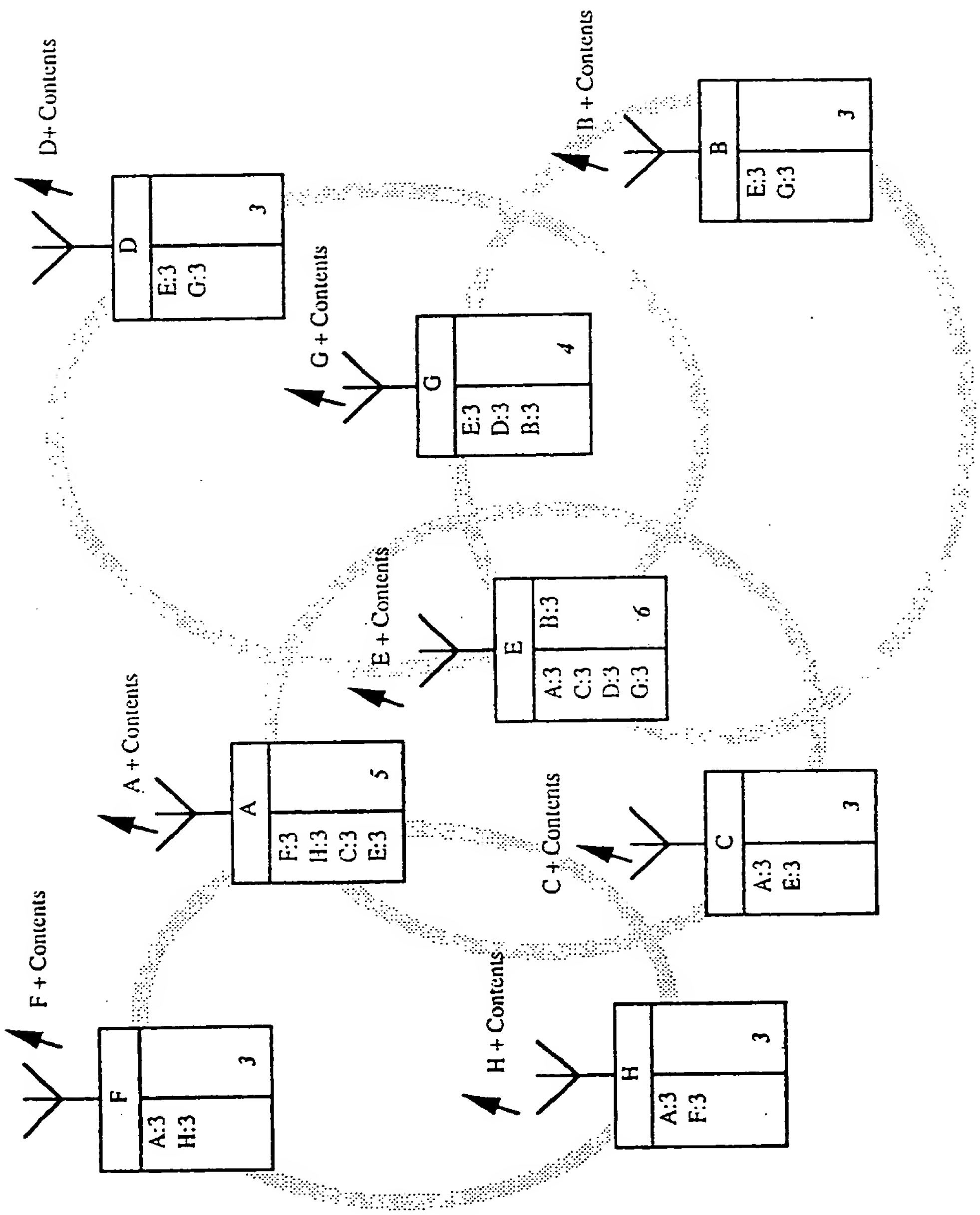

D+ Contents
D
E:3
G:3
3
B + Contents
B
E:3
G:3
3
G + Contents
G
E:3
D:3
B:3
4
E + Contents
E
A:3
C:3
D:3
G:3
B:3
6
A + Contents
A
F:3
H:3
C:3
E:3
5
C + Contents
C
A:3
E:3
3
F + Contents
F
A:3
H:3
3
H + Contents
H
A:3
F:3
3

Fig.22.

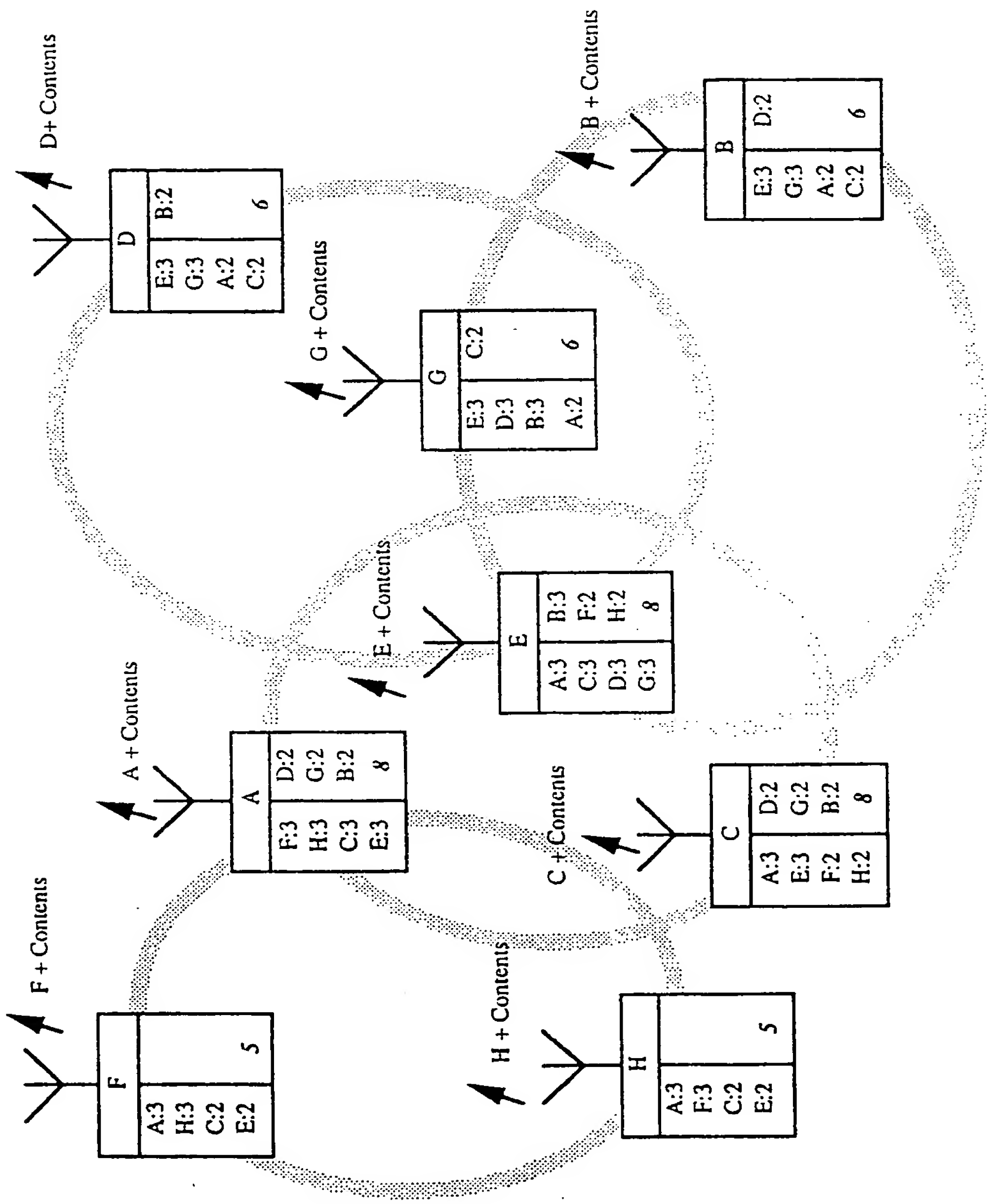

D+ Contents
D
E:3
G:3
A:2
C:2
B:2
6
B + Contents
B
E:3
G:3
A:2
C:2
D:2
6
G + Contents
G
E:3
D:3
B:3
A:2
C:2
6
E + Contents
E
A:3
C:3
D:3
G:3
B:3
F:2
H:2
8
A + Contents
A
F:3
H:3
C:3
E:3
D:2
G:2
B:2
8
C + Contents
C
A:3
E:3
F:2
H:2
D:2
G:2
B:2
8
F + Contents
F
A:3
H:3
C:2
E:2
5
H + Contents
H
A:3
F:3
C:2
E:2
5

Fig.23.

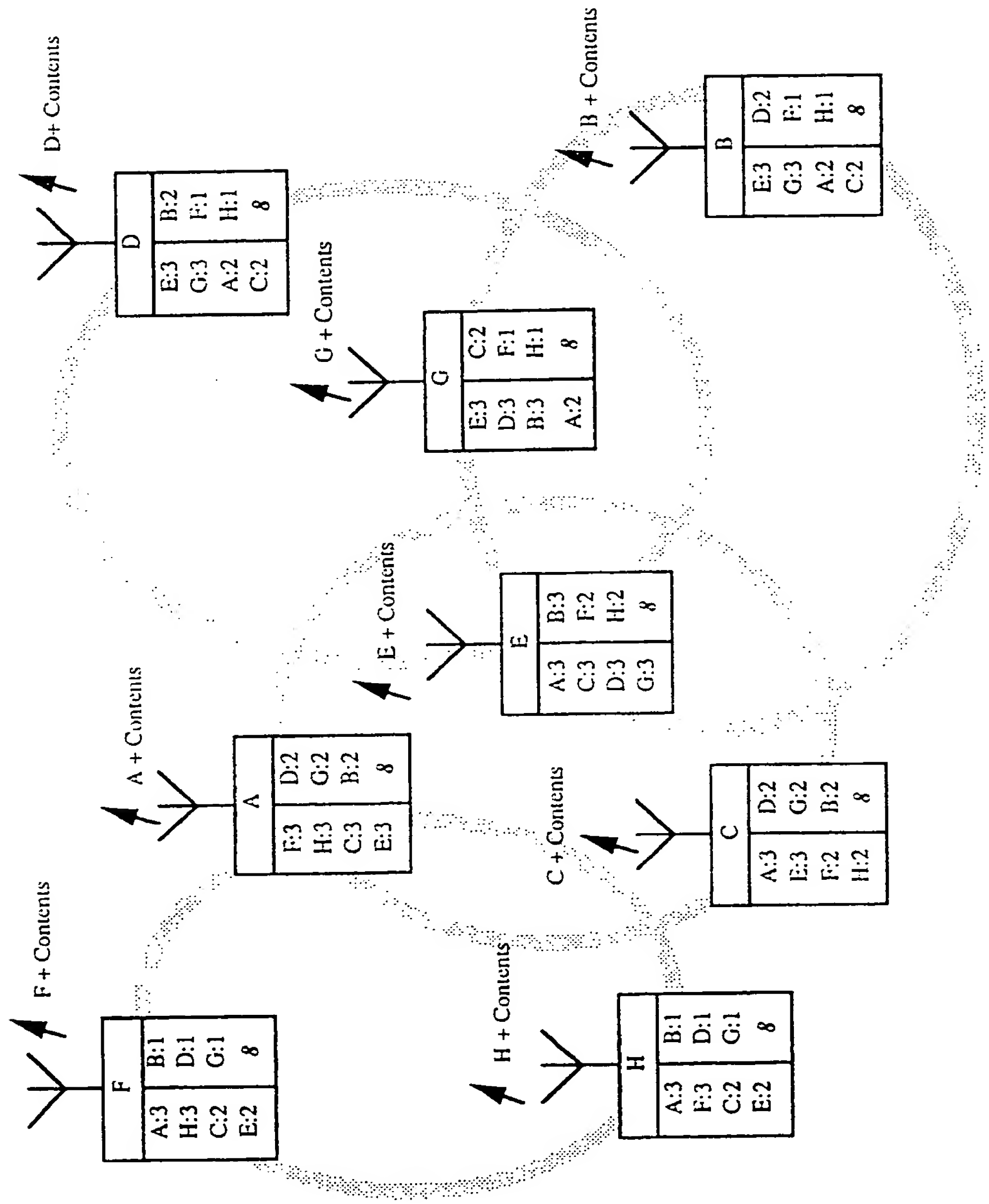

D+ Contents
D
E:3 G:3 A:2 C:2
B:2 F:1 H:1 ∞
B + Contents
B
E:3 G:3 A:2 C:2
D:2 F:1 H:1 ∞
G + Contents
G
E:3 D:3 B:3 A:2
C:2 F:1 H:1 ∞
E + Contents
E
A:3 C:3 D:3 G:3
B:3 F:2 H:2 ∞
A + Contents
A
F:3 H:3 C:3 E:3
D:2 G:2 B:2 ∞
C + Contents
C
A:3 E:3 F:2 H:2
D:2 G:2 B:2 ∞
F + Contents
F
A:3 H:3 C:2 E:2
B:1 D:1 G:1 ∞
H + Contents
H
A:3 F:3 C:2 E:2
B:1 D:1 G:1 ∞

Fig.24.

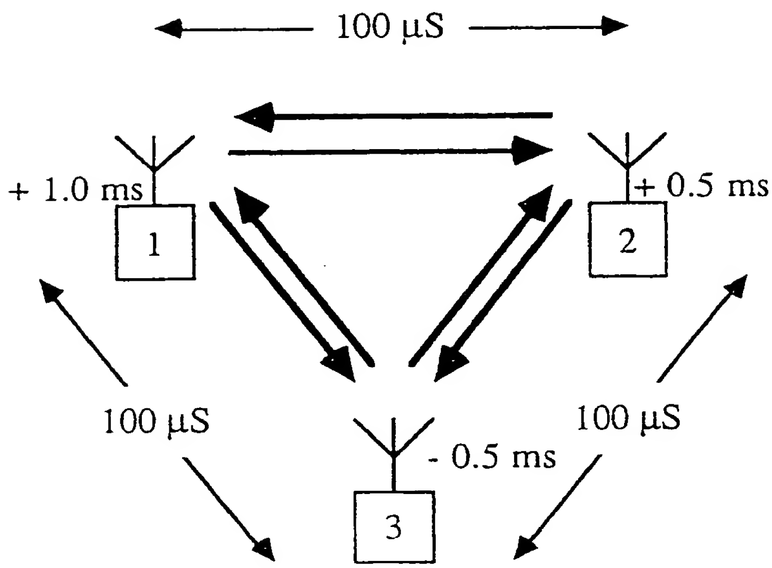

100 µS
+ 1.0 ms
1
+ 0.5 ms
2
100 µS
100 µS
- 0.5 ms
3

Fig. 25.

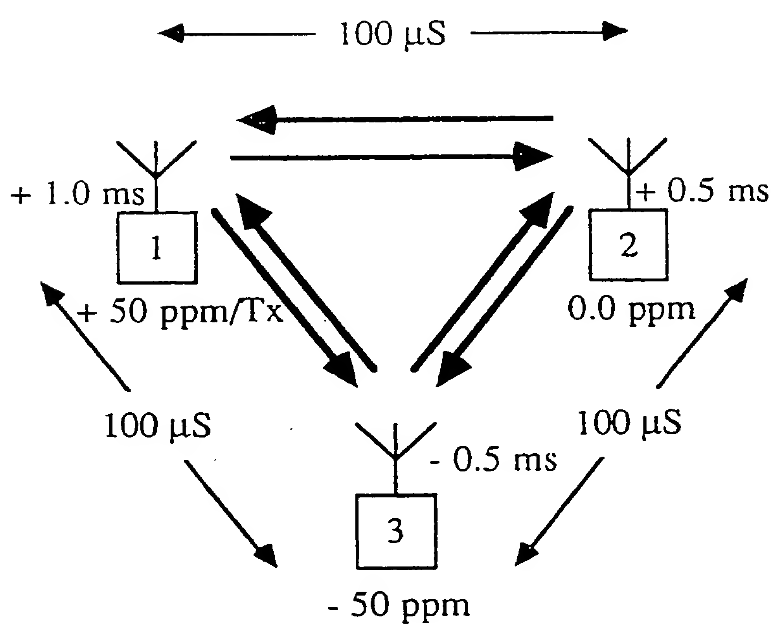

100 µS
+ 1.0 ms
1
+ 0.5 ms
2
+ 50 ppm/Tx
0.0 ppm
100 µS
100 µS
- 0.5 ms
3
- 50 ppm

Fig. 29.

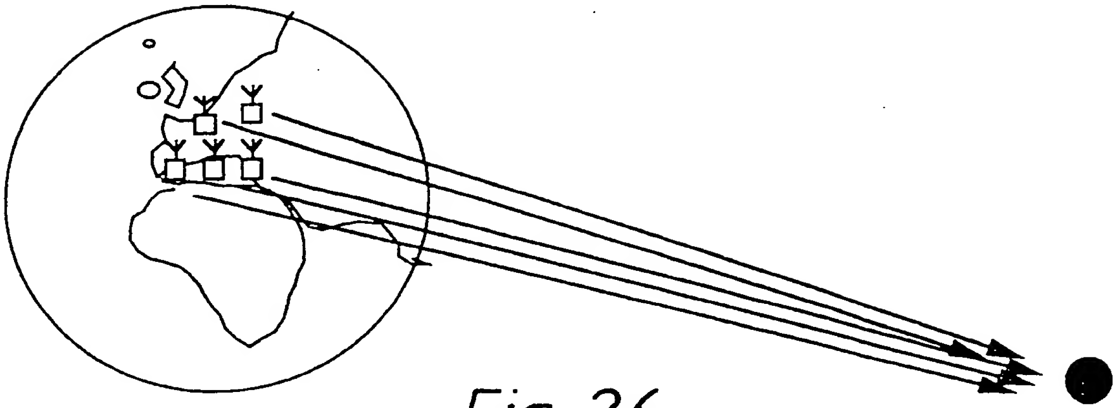


Fig. 26.

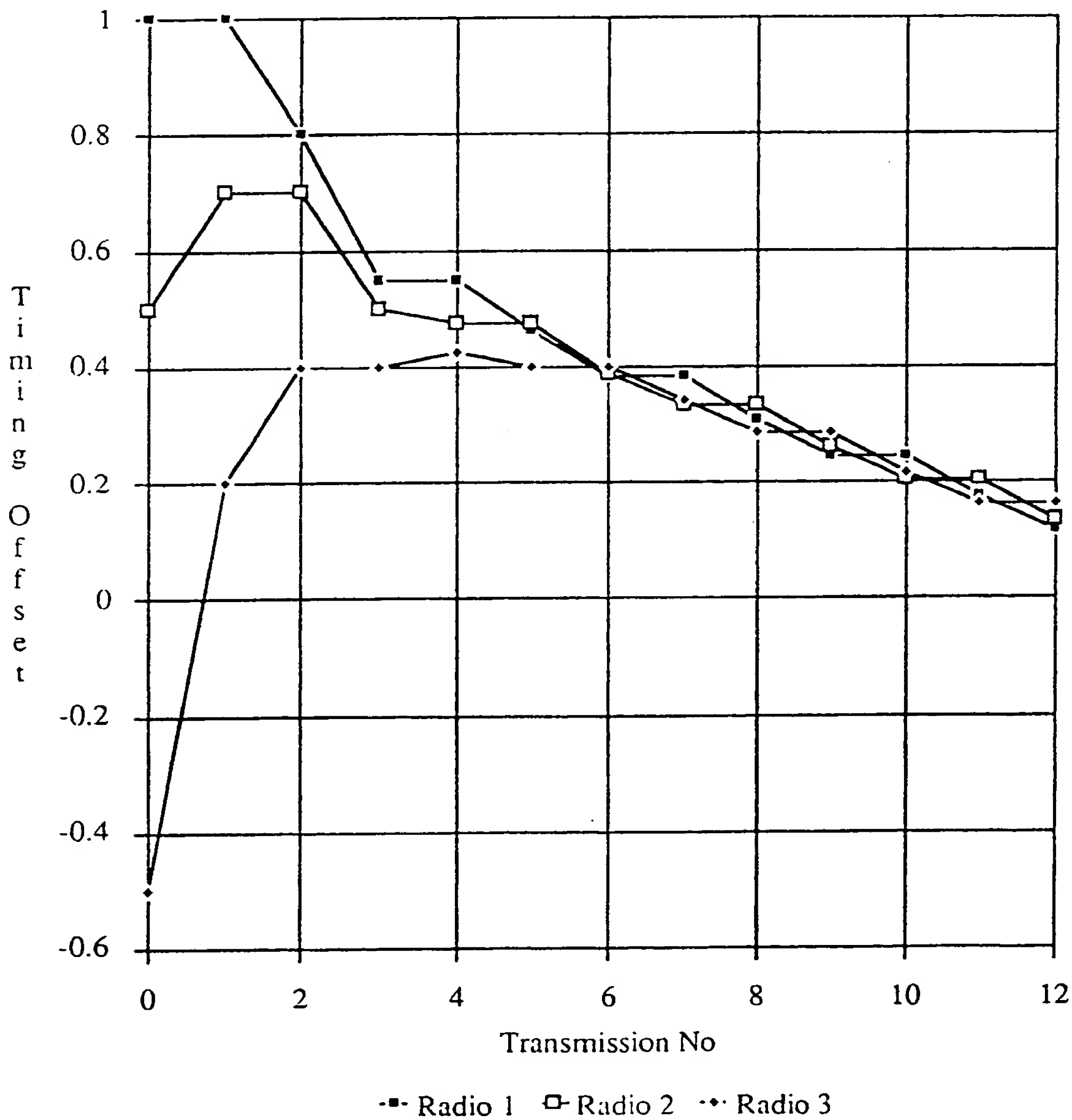

1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
Timing Offset
0
2
4
6
8
10
12
Transmission No
Radio 1
Radio 2
Radio 3

Fig. 27.

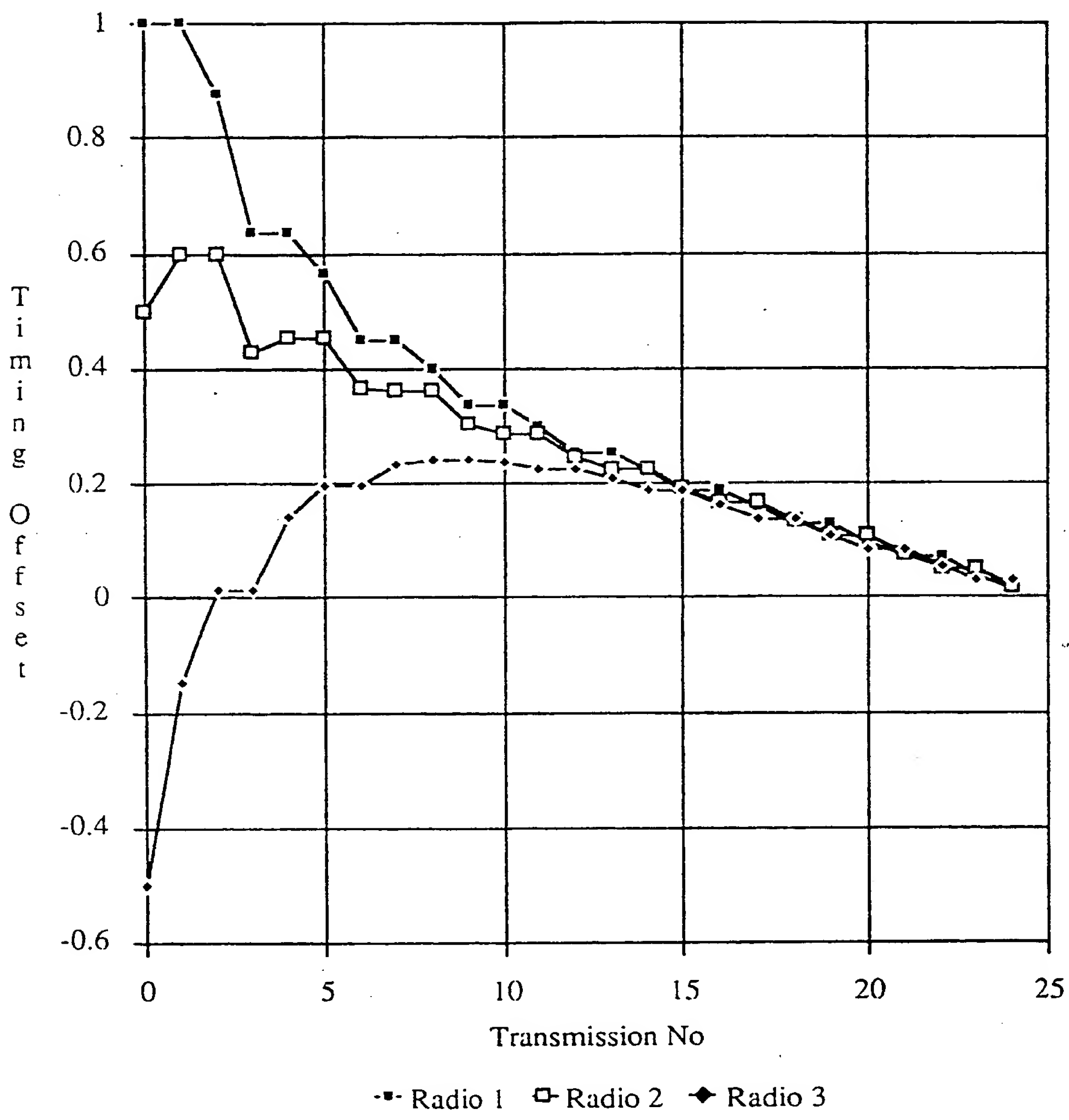

1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
Timing Offset
0
5
10
15
20
25
Transmission No
Radio 1
Radio 2
Radio 3

Fig. 28.

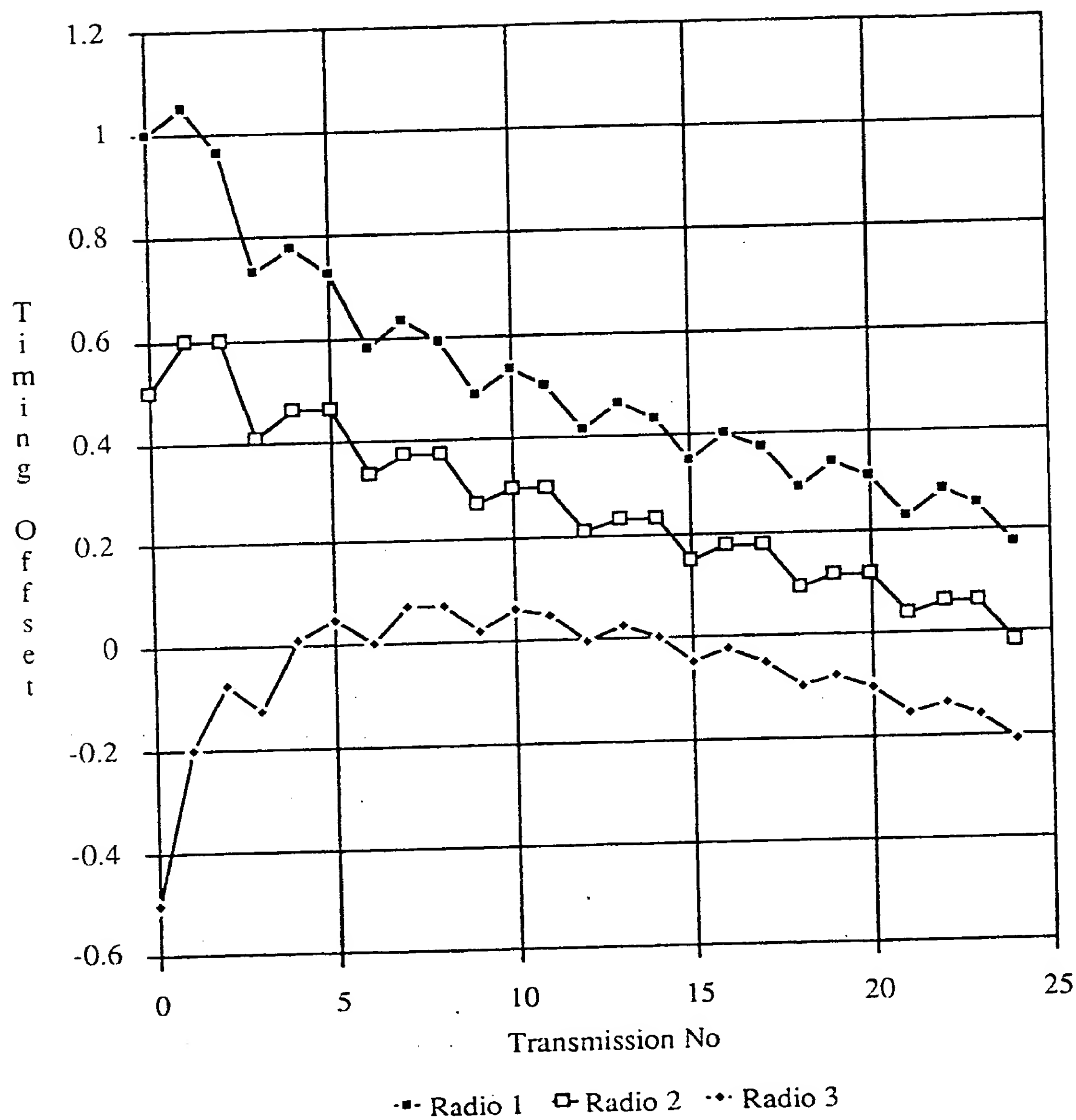

1.2
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
Timing Offset
0
5
10
15
20
25
Transmission No
Radio 1
Radio 2
Radio 3

Fig. 30.

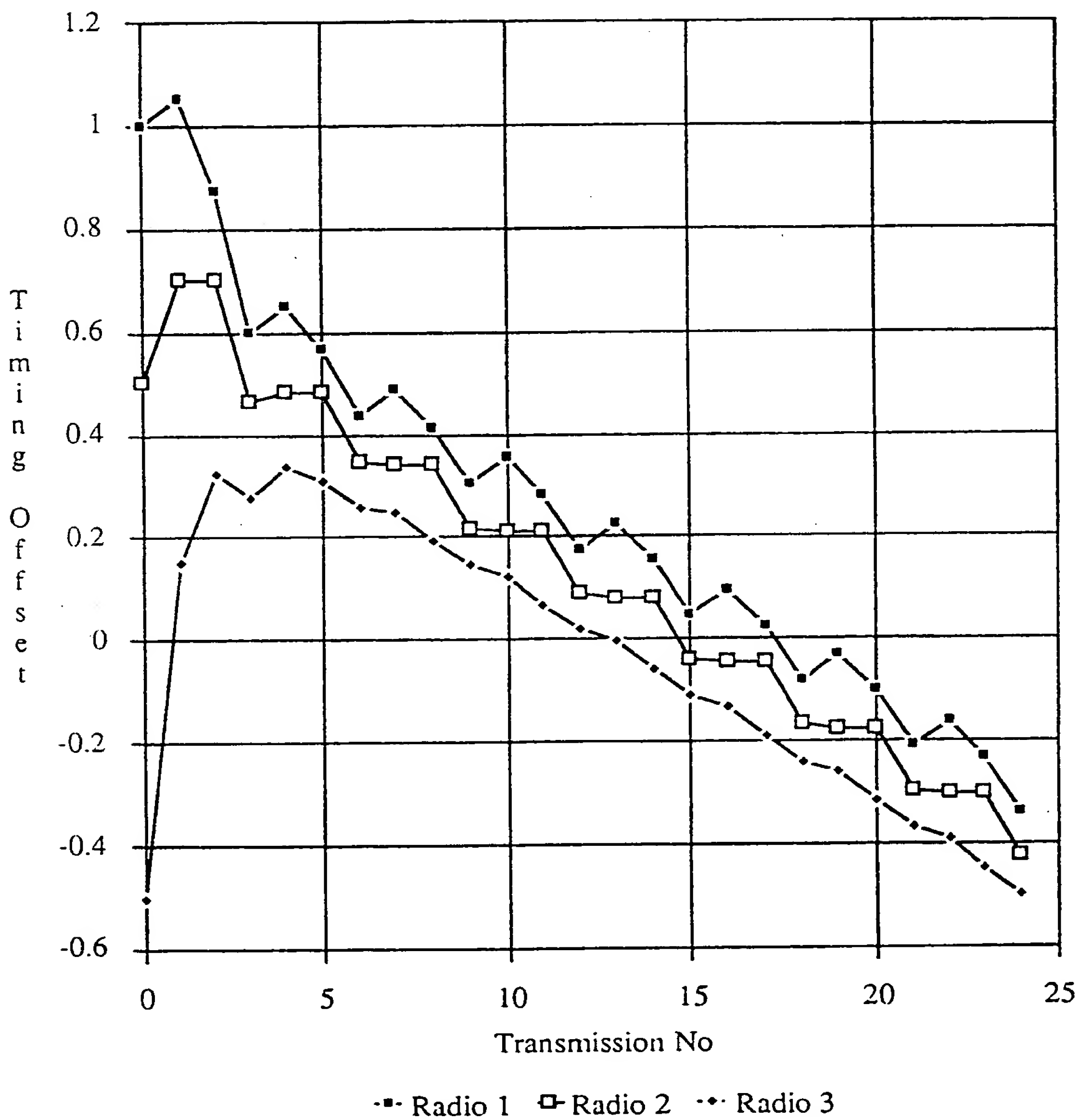
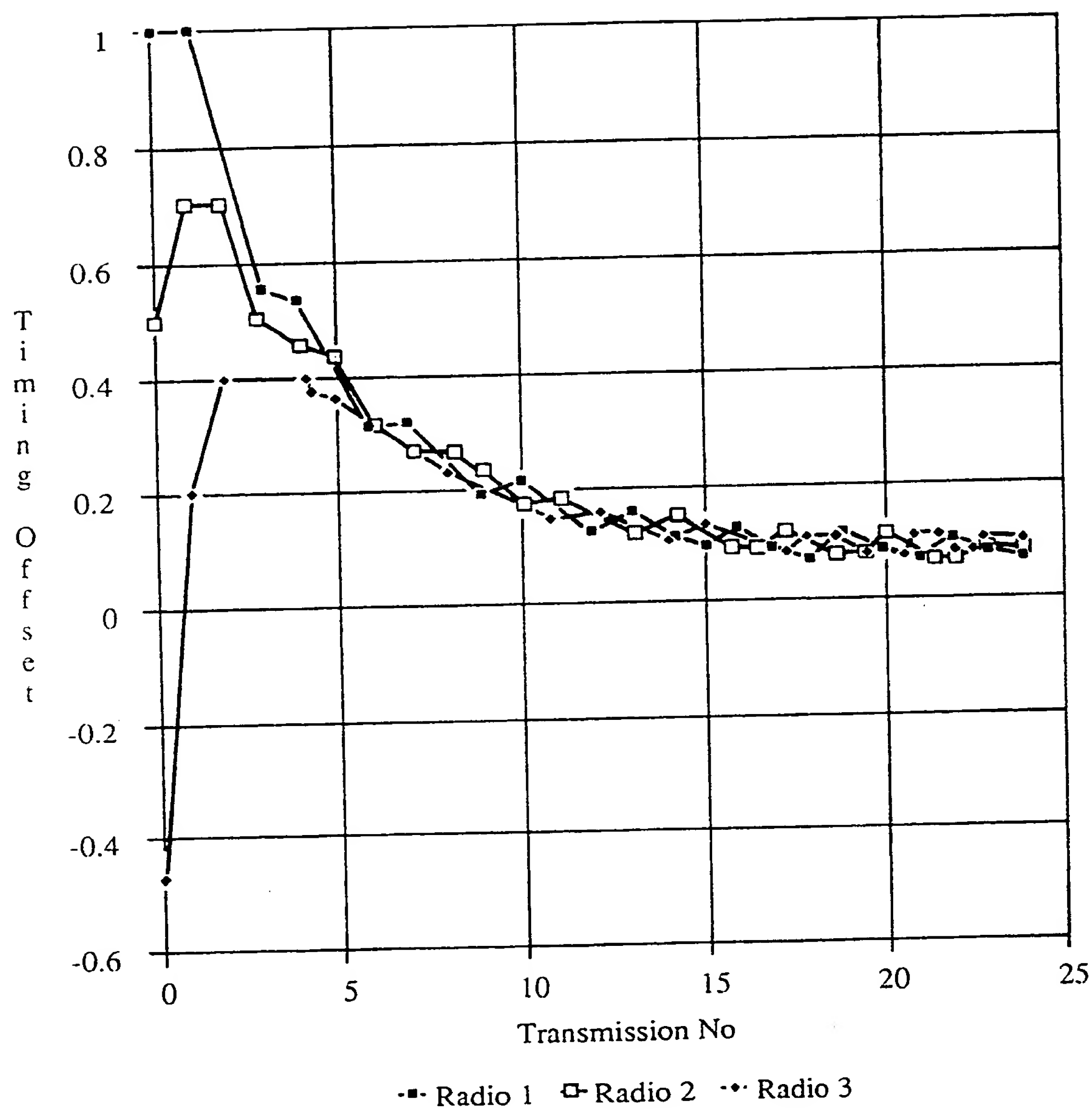

1.2
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
Timing Offset
0
5
10
15
20
25
Transmission No
Radio 1
Radio 2
Radio 3

Fig. 31.



1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
Timing Offset
0
5
10
15
20
25
Transmission No
Radio 1
Radio 2
Radio 3

Fig. 32.

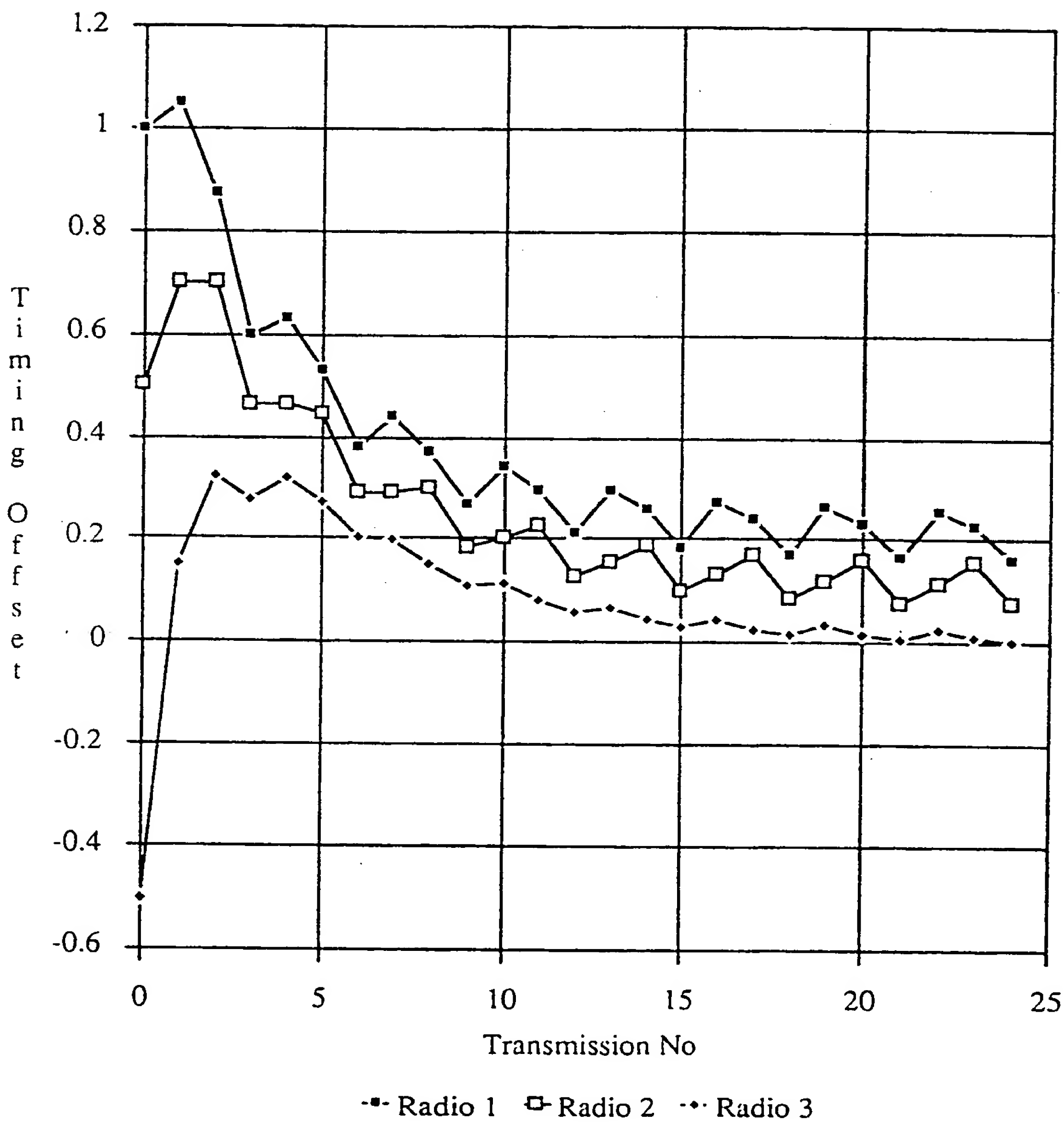

1.2
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
Timing Offset
0
5
10
15
20
25
Transmission No
Radio 1
Radio 2
Radio 3

Fig. 33.

2264842

WIDE AREA NODELESS DISTRIBUTED SYNCHRONISATION

This invention relates to radio communications.

A vital requirement for modern radio communication using frequency hopping is synchronisation. Recent work has shown that significant benefits may be obtained if entire networks or even networks of networks can be synchronised together. In particular, such synchronisation reduces system self interference and also provides inter network communications both for system management and long range user traffic.

One object of this invention is to solve the problem of synchronising a large deployment of radios without the use of any masters or additional equipments, when the radios may not all be within direct radio range and propagation delays may be significant compared with the required accuracy.

Another object of the invention is to provide novel handshake protocol for unsynchronised radios.

According to the present invention we provide a frequency hopping distributed (masterless) radio communication system comprising a plurality of radio transceivers each having a clock error prior to synchronisation not exceeding ± N seconds with respect to absolute time wherein the following operational conditions are arranged to obtain;

a) synchronisation transmissions are made by each radio transceiver at random time during an interval NM seconds where M is a constant, which interval is defined at the centre of successive contiguous window periods of (4+M)N seconds, one synchronisation

transmission per window, the synchronisation transmission and reception frequency being the same for all radio transceivers during each window period and being varied in accordance with a pseudo random sequence from window period to window period and each synchronisation transmission being arranged to include data indicative of the time of transmission within the interval NM seconds and data indicative of the origin of transmission;

b) all radio transceivers are arranged to respond to the reception of a synchronisation transmission with an acknowledgement transmission (unless already in synchronisation with the transceiver which made the synchronisation transmission) upon receipt of which acknowledgement transmission the radio transceiver which originated the synchronisation transmission is arranged to compute, in dependence upon the transmission and reception times of the acknowledgement transmission a common time which all radio transceivers involved are to assume; and,

c) the synchronisation transmission source transceiver is arranged to direct update command transmissions to each of the acknowledging radio transceivers containing instructions for timing data updates to bring the radio transceivers into synchronisation with a common timing so that all radio transceivers involved will make a timing adjustment thereby to achieve synchronisation.

An initial synchronisation transmission may contain data appointing the time and frequency of acknowledgment transmissions.

Alternatively an initial synchronisation transmission may contain data appointing the time for an acknowledgement transmission and the frequency for the transmission may be

determined according to that time from a pseudo random frequency hopping sequence.

An acknowledgment transmission may contain data appointing the time and frequency for an update command transmission.

An acknowledgement transmission may contain data appointing the time for an update command transmission and the frequency for transmission may be determined according to that time from the said pseudo random frequency hopping sequence.

Several different times may be appointed for multiple acknowledgements whereby each acknowledging radio transceiver chooses at random from those made available.

Each radio transceiver may include partition size data in the acknowledgement transmission, the said partition size data being derived in dependence upon the number of radios, including itself, already synchronised.

The originator of the synchronisation transmission, upon the receipt of the acknowledgement transmissions, may determine the common time on the basis of the partition size data contained in the acknowledgement transmission such that the common time is taken to be the time of a radio transceiver having the largest partition size.

The originator of the synchronisation transmission upon receipt of the acknowledgement transmission, may determine the common time on the basis of the partition size data contained in the acknowledgment transmission, and wherein when no single largest partition obtains an average of the times may be taken.

The common time may be taken as an average of the timings of the acknowledging radio transceivers weighted linearly according to the partition sizes of the respective radio transceivers.

Upon receipt of the update command transmission, an acknowledgement radio transceiver may be arranged to relay the update command transmission for receipt by other radio transceiver already synchronised to it.

Every radio transceiver receiving an initial update command transmission may be arranged to relay it at P random times.

Prior to transmission of any update commands, a synchronisation transmission originator, may determine an appointed time at which all updates should take place, which time may be encoded into the update commands to ensure substantially simultaneous synchronisation operations.

Increasingly, radio communications systems are making use of networking concepts to provide new facilities such as; integrated secure digital voice and data; addressed guaranteed message delivery over paths significantly greater than the basic radio paths; self managing communications resources with prioritisation and crisis management, and inter-net working, usually via gateways. Provisions of these facilities can be particularly problematical when ECCM capabilities need to be overlaid.

An attractive option for providing ECCM capabilities is the use of slow frequency hopping. It can be shown that frequency hopping with random non-orthogonal sequences can provide excellent spectral efficiency whilst removing the need for complex frequency management procedures. Provision of wide area synchronisation, to

better than the guard time between hops reduces system self interference whilst facilitating inter-net communications without the need for dedicated gateways. Essentially, communications between groups is possible providing they know each other's frequency hopping sequences and hold common time. Thus problems of frequency management, connectivities and ECCM would all be solved.

Provision of wide area synchronisation is traditionally provided by means of some broadcast time standard. However, this has the disadvantage that the time source represents a single point vulnerability. An alternative is to distribute some timing masters throughout the deployment. This still suffers from the problem of increased vulnerability and suffers the additional disadvantage that, unless a large number are deployed, communications will be required over paths significantly longer than those operated by standard radios. Clearly the most attractive solution is for radios to establish sync. in fully distributed and nodeless fashion. There are no points of particular vulnerability and no special additional equipment is required.

This invention is concerned especially to provide algorithms and protocols for Wide Area Nodeless Distributed Sync., and the following assumptions are made:-

In defining assumptions on clock accuracy a compromise is made between practicality and cost. A figure of ±1 ppm represents a reasonable compromise, since it is close to the state of the art for non-ovened TCXOs and involves a cost which is not excessive when viewed in relation to a complete radio equipment.

Another factor in relation to clocks, is the accuracy with which they are set initially. This is defined as $\pm$ N secs, henceforth and is generally reckoned to be of the order of a few seconds. Bearing this in mind it is also desirable that the range should not represent a "brick wall" beyond which synchronisation becomes impossible. The parameters assumed for frequency hopping are, that hopping is at H hops/sec.

Some embodiments of the invention will now be described by way of example with reference to the accompanying drawings in which; Figure 1 is a diagram showing transmission/reception hop timing,

Figure 2 is a diagram illustrating the synchronisation frame,

Figure 3 is a diagram showing the synchronisation frame with randomised transmission timing,

Figure 4 is a graph showing how timing error varies in accordance with the number of frames required to achieve synchronisation,

Figure 5 is a diagram showing acknowledgement timing and frequency,

Figure 6 is a diagram showing synchronisation between two radios,

Figure 7 is a diagram showing tuning between graphic and synchronisation channels,

Figure 8 is a diagram showing multiple slot watching,

Figure 9 is a diagram showing slot watching in a deployment of three radios,

Figure 10 is a table showing the watch state of radios shown in Figure 9,

Figure 11 is a diagram showing a deployment of three radios in a second slot watching state,

Figure 12 is a table appertaining to the watch state of radios shown in Figure 11,

Figure 13 is a diagram showing the deployment of three radios in a final slot watching state,

Figure 14 is a table showing the watch state of the radios shown in Figure 13,

Figure 15 is a diagram showing a slot watch algorithm,

Figure 16 is a graph showing watch algorithm performance,

Figure 17 is a diagram illustrating two radio slot sharing,

Figure 18 is a diagram showing four radios in a slot watching deadlock situation,

Figure 19 is a diagram showing two partitions or groups of radios in synchronisation conflict,

Figure 20 is a diagram showing a count updating process,

Figure 21 is a diagram of a deployment of radios illustrating a first stage radio counting process,

Figure 22 is a diagram which is similar to Figure 21 but which illustrates a second stage radio counting process,

Figure 23 is a diagram similar to Figure 21 and 22 which illustrates a third stage radio counting process,

Figure 24 is a diagram which illustrates progression of the counting process shown in Figure 23 and illustrating a fourth stage radio counting process,

Figure 25 is a diagram of a fine timing scenario with no clock errors,

Figure 26 is a diagram which illustrates common timing definition,

Figure 27 is a graph showing the performance of radios operating exponential updating,

Figure 28 is a graph showing an alternative exponential updating performance,

Figure 29 is a diagram illustrating a fine timing scenario with clock speed errors,

Figure 30 is a graph showing exponential updating in the presence of clock speed errors,

Figure 31 is a graph showing exponential updating with different clock speed errors,

Figure 32 is a graph showing exponential updating with predetermined compensation and;

Figure 33 is a graph showing exponential updating with specified errors and compensation.

The exact slotting of hops is as shown in Figure.1. It is assumed that the time taken for a receiver synthesiser to re-tune is insignificant compared with the guard period (this possibly implies the use of a pair of synthesisers). On this basis the receiver can readily hear any transmission which is up to $\pm 1/2$ G seconds in error, were G is the guard time.

Obviously the requirement is that the radios should synchronise as rapidly as possible, and in normal operation a pre-synchronised system will provide immediate service. After

significant periods of radio silence however, (e.g. several hours) the radios may drift out of sync. On this basis the network should re-establish sync. in a few seconds. In the case of synchronisation "from cold", however, a reasonable period can be allowed. For example, 15 minutes would probably not be unacceptable.

The fundamental purpose of synchronisation in a frequency hopping scenario is to facilitate user traffic communications. However, communications over the same medium must be employed to achieve synchronisation in the first place. Thus it is necessary to establish a mechanism for communications specifically dedicated to achieving synchronisation which does not itself rely on very good time accuracy but which nevertheless achieves the ECCM capability of frequency hopping.

One such approach is to establish a synchronisation channel which changes less frequently than the normal frequency hopping traffic channel, such that a radio transmitting on this channel can guarantee that the others in the deployment will be tuned to it at transmission time.

When a radio is unsynchronised it will listen permanently on the synchronisation channel, and time will be divided into frames during which the frequency remains unaltered, as shown in Figure 2.

This shows that a timing error of $\pm$ N secs for each radio results in a synchronisation frame period of 4N seconds. This arises because the transmitter could be up to 2N second ahead or behind the receiver. The use of such a frame on first appearance is satisfactory. However, consider the case where several radios have already synchronised but are continuing to send synchronisation messages

for the benefit of other unsynchronised radios. The synchronised radios will all transmit simultaneously, causing significant co-channel interference, often preventing reception.

This may be resolved by introducing a random element into the timing of the synchronisation transmissions. A useful compromise between randomness and extension of the frame period is $\pm$N secs = 1/2 MN where M is a constant and M = 2 in this case. The effect of this on the frame period is illustrated in Figure 3.

In fact, another advantage of the scheme just before described, is that it provides a gradual degradation in sync. performance for any radios whose clocks had, for any reason fallen outside the $\pm$N second window. Because the time of sync. transmission has a uniform distribution across $\pm$N seconds of the frame, a radio which is up to about 3N seconds in error will eventually succeed in receiving/transmitting the sync. message in the correct frame. Thus, for timing errors greater than +N seconds the worst case (worst case here means that the rest of the radios are synchronised N second in error in the other direction) the average time to achieve sync. will tend to increase according to the curve shown in Figure 4.

Considering now the next questions relating to the basic synchronisation transmissions which are; firstly who should make them? and, Secondly what should they do?

Synchronisation transmissions could be viewed as performing any of the following roles;

a) An opportunity provided by the initiator for others to synchronise.

b) A request by the initiator for synchronisation.

c) A combination of the above.

In role a) it could be argued that the initiator would need to view itself as being, in some sense, already synchronised. On this basis only those radios which have become synchronised to at least one other radio should make synchronisation transmissions. In this condition the system would never start up since all radios would remain silent, waiting for synchronisation transmissions which would never be made.

In role b) the same problem exists in a different form. Replies to a request for synchronisation can only logically be provided by radios which are in some sense already synchronised.

In a distributed sync. scenario the synchronisation might serve to connect two partitions, e.g. one partition containing 4 radios, the other containing 7. In this case "who is synchronising whom?" is a moot point. In fact the only logical role for synchronisation messages is that of role c), i.e. "it depends". On this basis, synchronisation messages are made by every radio in every frame.

The second question "What should it do?" is, in some sense, linked to the first, i.e. "It depends!". For example, a very simple approach would be to make the synchronisation transmissions constitute a basic "**JOIN ME**" command. This is clearly unacceptable because all informed groups would be regularly fragmented by individual radios coming into range on the periphery. Moreover, the system would be extremely vulnerable to spoofing of the record-and-reply kind and encryption would have no effect.

Clearly then, some form of 'handshake' is required to permit exchange of information and to validate the source and timing of the

synchronisation transmission. Thus the first response to a synchronisation transmission should be an acknowledgement transmission.

The first problem one meets with acknowledgement transmissions is, "How can it be sent to the originator of the synchronisation transmission, given that the two radios are frequency hopping and unsynchronised?". We do not wish to use the synchronisation channel for acknowledgement since this holds some potential for exploitation by spoofers after the initial synchronisation transmission.

Fortunately, it is possible to predict the correct time and frequency for transmission of the acknowledgement using information contained within the original synchronisation transmission, coupled with knowledge of the time at which this was received. Several procedures could be envisaged for this but the one proposed is as follows:-

1) As described above, time is divided into '6N' second frames. Moreover, every frame is sub-divided by the frequency hopping into a fixed number of slots (6NH for H hops/sec). Synchronisation messages contain, in digital form, the slot number in which they were transmitted.

2) When a radio receives a synchronisation message it notes its own slot number (i.e. determined by its own clock) - including any fractional component - when the message was received. By comparing the local slot number with that contained in the message, the radio can determine the time difference between the two radios.

3) In addition to the slot number the synchronisation message also contains a **slot delay** entry. This corresponds to the number of slots after sending the synchronisation message that the initiator expects to receive an acknowledgement.

4) The acknowledging radio can readily calculate from the above information, the time (with respect to its own clock) at which it should schedule the transmission to be received at the sync. source at the correct time.

The above takes no account of choice of frequency for the reply. It would be possible for the synchronisation message to contain, along with the slot delay entry, an entry specifying the frequency on which the acknowledgement is expected. However, this would be inefficient in the use of communications bandwidth, and is unnecessary. The proposed approach is for there to be a system wide synchronisation frequency hopping sequence independent of the normal traffic hopping sequence. Using the slot number plus the delay slot, the recipient of the sync. message can determine the absolute slot number on which the acknowledgement is expected and hence the particular frequency required, derived from the synchronisation frequency hopping sequence. This procedure is illustrated in Figure 5.

Referring now to Figure 5, the sync. source has randomly selected slot number 3456 in the current frame for transmission of its synchronisation message. Note that the sync. message has a length of one half slot and is transmitted twice. This ensures that whatever the timing of the recipient (assuming it is not always tuned to the sync. channel) the message can be received. In this case the

recipient receives the message during its own slot number 4429 and in fact notes that the exact reception time was 4429.5 (i.e. halfway through the slot). In determining the received slot number, the recipient references reception time back to the start of the message. The two repeats of the sync. message are labelled first and second so the recipient is able to determine the start time for the first of the transmissions even if it only received the second.

The sync. source randomly determined that it would listen for an acknowledgement 73 slots after transmission of the sync. message and placed this number into the sync. message. The recipient of the message adds this number to the local reception slot number to determine when to send the acknowledgement. The sync. transmission also contained the slot number in which it was sent (i.e. 3456). By adding the slot delay to this, the sync. recipient can determine in which slot number the sync. source expects to receive the acknowledgement.

Both sync. source and recipient run the same frequency sync. hopping algorithm in addition to the user traffic frequency hopping sequence. This generates frequencies for all slots but only a small number of them are actually used. The sync. source spends most of its time listening on the synchronisation channel (for sync. messages from other radios) but, in slot 3529 (=3456 + 73) it looks up the appropriate frequency for this slot from the frequency sync. hopping sequence and tunes to this. Similarly, the recipient of the sync. message, using knowledge of the slot number for reception at the sync. source can select the appropriate frequency for transmission. Note that it will usually be necessary for the acknowledgement to be

sent out of time with respect to the normal slotting of frequencies. The flexibility to do this means that it is not necessary for the acknowledgement transmission to be sent twice since it can be accurately time aligned with the receiver.

Note that recipients of synchronisation messages will respond with acknowledgement **only** if they are not already synchronised with the originator. This is tested by comparing the slot number contained within the message against the local slot number when the message was received. If the difference is less than the guard time specified for transmissions then the two are reckoned to be synchronised. Moreover, recipients of sync. messages will respond only to the **first** correctly received from a differently timed radio in that frame. Furthermore, any radio hearing a sync. message before it has transmitted its own for that frame will suppress its transmission.

In spite of the above constraints on acknowledgements, it is quite likely that several radios will all hear the same sync. transmission and legitimately acknowledge it. In this case, of course, their transmissions will interfere at the sync. source and none may be received. In order to avoid, or at least mitigate this effect, the sync. source in fact sends a number of options for slot delay for acknowledgement (4 were used in the model). The sync. source looks up the corresponding frequencies and listens on these at the appropriate times. In turn the acknowledging radios randomly select one of the slot delays for the acknowledgement. This does not always completely solve the problem in all cases and thus protocol extensions may be used as described later herein.

Having established a mechanism for returning acknowledgement messages to the sync. source, we must now consider what actions(s) these messages should perform. The same arguments apply as for sync. messages, to the effect that the acknowledgements cannot take the form of simple "**JOIN ME**" instruction, in this case because the sync. source could receive conflicting instructions from different acknowledgers. The opportunity to synchronise **all** of the acknowledgers to the sync. source would also be lost. It follows that any sync. source receiving acknowledgments should temporarily take on the role of arbiter for synchronisation of itself and all of the others it has heard.

In order to do this the sync. source must log reception of all acknowledgements and wait until all slots specified by slot delays in the sync. message have passed. At this time the sync. source may then process the acknowledgements in order to determine what synchronisation instructions to issue.

The necessary operations determine the required contents for the acknowledgment messages. Firstly, they must contain the slot number (including any fractional component), in which they were transmitted. This provides the time difference information necessary for the sync. source to compute appropriate time updating values to be sent in the synchronisation instruction message, hereafter referred to as update transmissions.

Secondly, a mechanism must be derived for correct reception of update messages. This is done in exactly the same way as for acknowledgement messages. In this case the acknowledging radio places a randomly determined slot delay into the acknowledgement

message for reception of any update message sent. The slot delay must have a minimum value such that no update transmission will be expected before the sync. source has received all acknowledgements.

Using the above information, the sync. source can determine what updates are necessary.

As described above, the purpose of update transmissions is to issue instructions for radios to alter their timing. Efficiency is maximised if all radios alter their clock timing more or less simultaneously (i.e. within a slot period). Thus, in general, update transmissions will specify, not only an amount to alter the local clock but also a slot number in which to do it.

The sync. source should ideally issue instructions for updates which result in the minimum number of radios having to make changes. However, where there is no obvious choice in this regard it will calculate updates to alter the times of all equipments involved (including itself) to the average of their previous values. If we consider the case of just two radios, i.e. the sync. source and one acknowledger, then if one is 3 seconds fast and the other 1 second slow, they will synchronise to 1 second fast. The full synchronisation procedure for just two radios is illustrated in Figure 6.

It is important to note that, if radio 'B' had happened to transmit its sync. message before radio 'A' then the procedure would have been inverted. However, the result would have been the same since 'B' would have acted upon the same information.

We define a partition as any group of radios, less than the totality, which are in synchronisation. In general two or more radios

in synchronisation are viewed as a partition - in fact a lone radio can be viewed more generally as a partition of one. The mechanism for synchronisation of partitions is essentially as described above but with a few extensions.

The most important point with regard to partition synchronisation is that whenever a radio engages in a synchronisation dialogue with another it does so as a representative of its partition. Thus a radio may not simply synchronise itself to another if it is already a member of a partition. Either its whole partition moves with it or it does not move at all.

The above raises a vital issue for system stability. It is, in principle, possible for a single radio to initiate a time change for a 100 others. It is vital that the algorithms in the system will not permit this to happen unless it is appropriate. For example, it might be appropriate if the representative of the 100 radios had heard a representative of a partition of 200 radios. In fact this example gives the clue to resolving such issues. If every acknowledgement transmission contains the number of radios in (or perceived to be in) the representative's partition, then the sync. source can make intelligent decisions as to which radios should update and which should not. Counting the number of radios in a partition of arbitrary size is not trivial and is dealt with in another patent application, however for the purposes of this discussion the availability of an effective means will be assumed.

The procedure for processing acknowledgements is somewhat more complex than originally presented. The procedure is as follows:-

1) It is quite likely that acknowledgements may be received from more than one representative of the same partition. In this case only one of these is processed although information from both may be used (e.g. if they have different values for the number in the partition the higher will be taken).

2) The number of radios in each of the partitions represented by the sync. source and the acknowledgement sources are compared. If any exceeds all the others (taken separately) then the timing of this one will become the timing for the new, large, partition about to be formed. In this way it will be unnecessary for updates to take place within the largest partition. If however, there is no winner, then all partitions will update to a new time determined by the average, weighted according to number in partition, of all partitions represented.

3) The sync. source determines a time, a little into the future, at which the updates should take place and communicates this to all partitions along with the necessary time change in update transmissions. The reason for the delay in performing the updates is to permit the representative of each partition to inform the other members of their partitions of the necessary changes.

As mentioned earlier, it is the responsibility of the radio receiving the sync. update transmission to ensure that other members of its partition follow suit. It does this making transmissions on the normal communications channel, i.e. the normal frequency hopping sequence. The question of how the members of a partition can simultaneously listen on the communications frequency channel and on the framed synchronisation channel is presently

glossed over but is dealt with in another patent application. The important point to observe here is that communications between the members of a synchronised partition is straightforward.

Essentially, the requirement is for the update to be broadcast across the partition. The problem of broadcast across a non fully connected network (i.e. one requiring retransmissions) is notoriously difficult. Fortunately this only applies in cases where the requirement is for guaranteed delivery to all destinations. In this case it is sufficient to reach a high percentage of destinations. Those missed can be "mopped up" in subsequent synchronisation frames.

The procedure for intra partition update transmissions is relatively straightforward. The initial recipient of the update from the sync. source will repeat it (with a few minor alterations) several times at pseudo random intervals. Any radio within the partition hearing the update will also repeat it the same number of times. Thus the information will propagate across the partition. The number of repeats used is a compromise between the time taken out of normal traffic communications and reliability of update communications.

The foregoing description has described in some detail, the operation of the synchronisation procedure. Nevertheless, there are several important operations which have been glossed over, and these will now be considered in more detail.

As just before described, groups of radios become synchronised by making synchronisation transmissions. These are made on the synchronisation channel which remains fixed for a one minute frame. Radios which have not synchronised to any other, listen permanently

on the synchronisation channel. However, once they have become synchronised they must use a traffic (frequency hopping) channel to facilitate normal communications with the other members of the synchronised partition.

In principle this would be entirely satisfactory provided no other radios could be required to join the partition. However, in practice, new radios may join the partition at any time (late entrants). Moreover, other partitions formed in geographically remote locations may move into vicinity and require synchronisation. In practice there is virtually no condition under which the members of a partition can be certain that no further synchronisation will be required.

Thus it is necessary, at all times for a watch to be maintained on the synchronisation channel - even during normal traffic communications. Conceptually the simplest way to achieve this would be with a second receiver. Whilst this might not be too unattractive (it would not need to be agile, requiring only one re-tune every minute), it was considered worthwhile exploring less hardware intensive alternatives.

If the transmission format of the normal traffic communications were redundant it would be possible to use some of the reception time during which a radio transceiver is tuned to the synchronisation channel without loss of communications. For example, a coding scheme whereby loss of up to 33% (i.e. 1/Q where Q is 3) of transmissions would be acceptable could be envisaged. Thus a receiver could tune to the traffic channel for two slots out of every

three and to the synchronisation channel for the other. This is illustrated in Figure 7.

In the normal case the frequency hopping algorithm determines the tuned frequency F_N for all slots. When synchronisation channel watching is applied, the frequency hopping algorithm determines tuned frequency for two out of every three slots, the third being tuned to the synchronisation channel.

The above is useful but offers only a one in three chance of hearing a synchronisation channel. However, where a deployment of several radios exists it is, in principle, possible to establish a permanent watch on the synchronisation channel by assigning some to watching in slot 0, some in slot 1 and some in slot 2, as shown in Figure 8.

This shows the three possible assignments for watching. Note that in each, two thirds of communications slots are watched and that together the radios maintain a permanent watch on the synchronisation channel.

For a deployment of radios, therefore, what is needed is an algorithm which tends to ensure that the watch on the three slots is distributed as evenly as possible over the deployment.

The watching of the synchronisation channel is established using a nodeless algorithm. It is explained most simply with respect to the case of three radios attempting to set up a full watch together. Consider the the deployment of Figure 9. Each radio sends a transmission indicating its current slot watch state - in this case 0 for all radios. Thus all radios will hold a table as shown in Figure 10.

Since, in this case (the worst case) all radios are watching on the same slot, the system state is **SAME.** In this case all radios make a random choice of the slot number to use in the next synchronisation frame. For example, as shown in Figure 11 and in this case, each of the radios will hold a table as shown in Figure 12. The state is now, **2 SAME.** When the system is in this state, the radio whose slot number is different from the others retains that slot number. The others choose randomly from the other options (in this case 0&2). One result might be as shown in Figure 13.

Now each radio holds the table as shown in Figure 14 and the state is **ALL DIFFERENT** so no further action is taken by any of the radios.

Of course, bearing in mind the randomness of the operations involved the route to the final sate could have taken several directions. Nevertheless only the three states described are possible and convergence to the final state is good. In fact there is conceptually another state. This is the initial state and is the point at which no decisions about slot numbers have been made at all. The obvious thing to do at this time is to make a random selection of slot number. Thus the initial state is equivalent to the **ALL SAME** in the way it is handled. However, the existence of this initial state improves performance since there is already a reasonable probability of jumping to the final state from initialisation. The overall operation and performance of the algorithm is shown in the state diagram of Figure 15.

This can be used to analyse the probability of being in any given state following a number of iterations. The result of this

exercise is shown in Figure 16, wherein the curve shows that the convergence of the system to the desired state is quite rapid in spite of the non deterministic nature of the process.

As already mentioned, the case of three radios is rather simpler than the more general case. The algorithms have extensions to cater for these other cases as follows:-

Firstly consider the case of only two radios. It is still necessary for them to communicate but they should also maintain a complete watch of the sync. channel since they need to join a large partition as quickly as possible. The solution is to aim for slot sharing as shown in Figure 17.

In this case both radios watch two slots i.e. each radio watches for a period S/Q where S = 2 and Q = 3 of the time available for signal reception, where S is the smallest integer which is greater than or equal to Q divided by the number of radios R in a group or partition, the aim being to avoid them both watching the same pair. It is not difficult to see how an adaptation of the algorithm described above will achieve this.

When there are more than three radios in the network there are potentially several approaches to establishing a complete watch. Following experimentation with complex algorithms involving up to 6 or 7 radios it was found that the most effective solution was based on the simple approach of applying the basic algorithm to each radio and its two nearest neighbours (i.e. those received with the greatest signal strength). This method is generally satisfactory but can be subject to deadlocks, see Figure 18.

Consider the states perceived by each of the radios:-

Radio A sees B & C assumes state is 1 **DIFFERENT**

Radio B sees A & D assumes state is 1 **DIFFERENT**

Radio C sees A & D assumes state is 1 **DIFFERENT**

Radio D sees B & C assumes state is 1 **DIFFERENT**

In every case, each radio assumes it is the one to remain on the same slot. Thus slot 1 is never visited.

This condition is resolved by a slight modification to the watching algorithm wherein each radio considers its 3 nearest neighbours and uses information from a random subset of 2 radios.

In the aforegoing discussion it has been assumed that the situation is described by a number, n, of radios suddenly being deployed/activated in sync. and needing to establish optimum slot watching. This simplification was useful to explain the basic tenets of the algorithm. However, the reality is that the number of radios in a partition gradually grows from unity and that the slot sharing algorithm must operate near optimally during this growth. Inevitably this growth involves a large number of special cases which ideally are handled by specific tests and corresponding actions. Within the model, a rule base has been constructed consisting of about 30 rules for these specific cases. In many situations it is possible to avoid or minimise the need for randomness in the choices made because every radio is able to exploit the uniqueness of its situation.

Conflicts can arise in the operation of the sync. algorithm where different representatives of two partitions attempt to initiate synchronisation simultaneously. Consider Figure 19.

Two partitions are shown, 1 & 2. Radios A and B perform a handshake for establishing synchronisation at a similar time to radios C and D. Unless they both make the same decision as to which partition should move and by how much then a conflict will arise.

The most effective way to avoid a conflict in the timing updates between radios is for all members of each partition to be aware of the number of radios in its partition. Once established this permits a majority decision to made upon which partition should update, i.e. the partition with the larger number of radios will not update. This is better than making a random choice because it resolves conflicts and because it ensures that the minimum number of radios make updates.

For networks of arbitrary size, the determination of the number of radios in a partition is a non trivial problem. It is important to retain a nodeless approach to the process to avoid the generation, even temporarily, of key nodes within the network. Moreover, very large networks may require considerable control traffic associated with the counting process.

The approach adopted for counting radios in the partition is fully nodeless and results in a time to count for the partition which increases with the number of radios in the network but which does not require increasing control traffic levels.

The principle is as follows:-

Radios make transmissions within the partition for the purpose of conveying radio count information. Using the information contained in these transmissions radios maintain a list of other radios which they know to be members of the partition. The radios use this

list to construct further such transmissions, broadcasting the list (or, more generally, parts of it) to other radios. The process is illustrated in Figure 20. The initial generation of radio lists is seeded by each transmission including the name of its source. Initially, each radio's list will contain only the names of its immediate neighbours. However, in due course, as a radio's neighbours start to send transmissions including the names of their neighbours, it will obtain entries for increasingly remote equipments.

As described so far, the algorithm is unable to remove entries for radios which have, for any reason, left the partition. In principle, transmission containing such entries could "bounce around the network for ever". To cater for this case each entry in the list has an associated "lifetime" number. This is set to an initial, system determined, value when the name is first received as the identity of the source of a transmission. The lifetime number is transmitted with the equipment name in all count update transmissions. At the start of every synchronisation frame, the lifetime number of all entries in each equipment's radio list are decremented and any entries in which the number has reached zero removed. Thus, for an initial lifetime value of 3, an equipment's entry will remain in the system for a maximum of three frames after it has gone.

The choice of value for the initial lifetime is a compromise. Small values will ensure rapid updating following the loss of an equipment. However, for widely spaced networks requiring many rebroadcast steps, if the lifetime entry is too small then some entries may be removed before they have reached the extremities of the network. The effect of this is that radios in the centre of the

deployment retain a more accurate count of the number of radios than those at the edge. A refinement to the algorithm in which the initial lifetime parameter is made adaptive on the state of the network is under consideration. This would make the counting of radios in small networks more responsive to lost radios than in large networks.

An additional refinement to the algorithm is necessary. In the above form, the length of the individual transmissions would need to grow without limit with the size of the network. For example, in a deployment of 100 radios, every transmission would contain the name of all one hundred. This is undesirable since the control traffic would begin to encroach on the space available for user traffic. Moreover, it is unnecessary since radios generally have several neighbours and can maintain their radio count on the basis of information received in transmission from all of them.

The approach adopted is to make each transmission contain a random subset of those held in the radio list whenever it contains more entries than can be sent in a single transmission. Figure 21 shows a deployment of radios and their associated tables. The circles indicate the regions of connectivity, i.e. the all informed group areas. In the initial stage, no information is contained in the tables. Thus the only information transmitted is the identity of the source. The entry in the lower right had corner of each table is the current perceived number of radios in the network.

In fact, within any frame, a count transmission is sent on each of the three watch slots (not necessarily contiguously) to ensure that all immediate neighbours have a chance of hearing at least two of

them. In order to maximise efficiency, it is arranged that the random subsets of the radio list contained on each of the transmissions are non overlapping.

Finally, to illustrate the process, consider Figures 21 to 24.

In Figure 22, we see the deployment after one transmission frame. There is now an entry for each of the immediate neighbours of each radio. Note that the lifetime entry associated with each radio in the table is set to 3.

In Figure 23, we have the position after a further frame. Note that the lifetime counter for the immediate neighbours is still 3. This is because it has been reset following reception from the neighbours. However, at the start of the frame it had been decremented to 2 for all entries and this was the figure which applied when each radio sent its contents. Thus the entries in the table for "neighbour's neighbours" have the value 2 for their lifetime entry.

Note that the radios near the centre of the deployment already have the correct count for the radios in the network.

Figure 24 shows the network in the steady state condition. Following the arguments for Figure 23, it is apparent why the lifetime counter for some entries has fallen to 1. Additional radios on the extremes of the network would not have accurate counts on this basis.

Note that the example does not illustrate the case where only a subset of entries can be transmitted. This makes the operation of the algorithm somewhat undeterministic and therefore difficult to illustrate. However, modelling has shown it to be effective.

One important object of the present invention is to achieve system wide sync. in a totally nodeless fashion (i.e. without any master(s) in the network(s)). Achieving this is a particularly interesting problem in the area of handling small timing errors and achieving a global mean.

The obvious approach is for every radio to update its timing to that of all others. However, if this is done then individual new radios just joining a large partition of radios have the power to pull other radios away from the median in a way which does not make for stable small errors.

A compromise approach is proposed in which, rather than removing the timing error between a received signal and the receiver clock, the error is reduced to a fraction 1/V of its previous value where V is a constant greater than one, chosen to provide a compromise between stability and speed of response. The value of V is typically between two and four, this approach is referred to as exponential updating.

As an example consider the simple scenario illustrated in Figure 25. The three radios have mutual propagation delays of 100 μ secs and timing errors of +1,+0.5 and -0.5 ms with respect to some arbitrary "perfect" time. The presence of the propagation delays throws into question the meaning of common time. This can be viewed in terms of the diagram shown in Figure 26.

The definition of common time is as follows:- A set of radios is in sync. if, should they elect to transmit at the same time according to their local clocks, the transmissions will be received simultaneously

at some point in space perpendicularly above the centroid of the deployment and at some arbitrarily large distance.

Thus the concept of common timing is meaningful. The deployment of radios operating exponential updating is illustrated by the performance shown in Figure 27. The effect is that there is a hunting of radio timing about the area median value as each of the radios attempts to adjust its time towards the timing of the signal received, which is later than the time at which the signal was transmitted. Moreover, the continuing updating results in a general trend of increasing delays with time. Thus, on average, the clocks tend to run slow. Both of these effects can be ameliorated by reducing the updating coefficient. Figure 28 shows the effect of using a figure of 25% (Note that the horizontal axis is compressed and note that the errors between the radios is significantly reduced).

In the previous case it was assumed that the clocks in the radios started with errors but that they were running accurately. We now consider the case where there is an error in the clock speed. This illustrated in Figure 29 which shows the previous case but with specified clock speed errors.

Note that the clock speed errors are specified per transmission. This simplifies the analysis and is reasonable on the basis that transmissions are made at regular intervals. Figure 30 shows the effect of clock errors on the timing errors for 25% update coefficient (i.e. each radio reduces its error from the received transmission by 25%).

The average difference between radio timings is now approximately 200 μs, i.e. 4 times the clock speed error per

transmission. This is intuitively understandable on the basis that the steady state is reached when the error increase between transmissions is equal and opposite to the update made on receipt of the transmission. If the error is 200 μs, on average then it will be a little over this value just before receipt of a transmission and will be reduced by 25% on receipt of the transmission - i.e. by 50 μs. The clock error during the next interval will make up this 50 μs. Thus we see that the steady state error is multiplied by the reciprocal of the update coefficient - in this case 4. In order to reduce the errors it is desirable to increase the update coefficient as far as possible. The effect of increasing it to 50% is shown in Figure 31.

Here we see that the error between radios has reduced to 100 μs but that the rate of drift has increased back to that of Figure 27.

Drift is not a problem per se within a synchronised network but rapidly becomes a difficulty as soon as a radio leaves the network temporarily. In this case the absentee will soon stop drifting with the network and will rapidly become unsynchronised. It is desirable, therefore to eliminate the systematic drift if at all possible.

This section presents a novel algorithm for eliminating drift due to propagation delays between radios. Every radio is able to keep a log of the updates which it has made in the previous frame. This information may be used to compensate the clock speed of the equipments for drifts due to propagation delays. If an equipment uses the information which it holds locally to perform the updating then the effect is self defeating - it will compensate for updates caused by propagation delays and caused by genuine clock speed error. The effect is that such compensation causes the compensated

equipment to drift away from the others as though no fine synchronisation procedure were in operation.

However, if the equipments convey, in their transmission, their log of updates in the previous frame then the equipments can compensate for the updates using a spatial average of the updates performed. If all radios do this then this average will relate only to the effect due to propagation delays. In addition exponential time averaging of the spatial average may be applied in order to smooth out randomness in transmission times etc.

Figure 32 shows the effect of applying the compensation algorithm to the three radio deployment of Figure 25 (i.e. with no clock speed errors).

It will be noted that the curves of timing slope upwards toward horizontal as the information is accumulated and the time averaging takes effect (a time averaging coefficient of 50% is applied). Figure 33 illustrates the effect of compensation on the three radios deployment with clock speed errors (as per Figure 31).

In this case the steady state error between radios is modest and the long term drift has been eliminated. It is noteworthy that the drift of the radios is zero because the average of the clock speed errors in the example is zero. In a general case the drift will be equal to the average clock speed error of the radios in the network. This is advantageous because it means that the drift of the network will tend to be smaller than the drift of an individual radio. Thus the time for an absent radio to drift out of sync. will be dependent largely upon its clock speed error without the need to consider the drift of the network to the same extent.

CLAIMS

1. A frequency hopping distributed (masterless) radio communication system comprising a plurality of radio transceivers each having a clock error prior to synchronisation not exceeding ± seconds with respect to absolute time wherein the following operational conditions are arranged to obtain;

a) synchronisation transmissions are made by each radio transceiver at random time during an interval NM seconds which interval is defined at the centre of successive contiguous window periods of (4+M)N seconds, one synchronisation transmission per window, the synchronisation transmission and reception frequency being the same for all radio transceivers during each window period and being varied in accordance with a pseudo random sequence from window period to window period and each synchronisation transmission being arranged to include data indicative of the time of transmission within the interval NX seconds and data indicative of the origin of transmission;

b) all radio transceivers are arranged to respond to the reception of a synchronisation transmission with an acknowledgement transmission (unless already in synchronisation with the transceiver which made the synchronisation transmission) upon receipt of which acknowledgement transmission the radio transceiver which originated the synchronisation transmission is arranged to compute, in dependence upon the transmission and reception times of the acknowledgement transmission a common time which all radio transceivers involved are to assume; and,

c) the synchronisation transmission source transceiver is arranged to direct update command transmissions to each of the acknowledging radio transceivers containing instructions for timing data updates to bring the radio transceivers into synchronisation with a common timing so that all radio transceivers involved will make a timing adjustment thereby to achieve synchronisation.

2. A system as claimed in Claim 1, wherein an initial synchronisation transmission contains data appointing the time and frequency of acknowledgment transmissions.

3. A system as claimed in Claim 1, wherein an initial synchronisation transmission contains data appointing the time for an acknowledgement transmission and wherein the frequency for the transmission is determined according to that time from a pseudo random frequency hopping sequence.

4. A system as claimed in any preceding claim, wherein an acknowledgment transmission contains data appointing the time and frequency for an update command transmission.

5. A system as claimed in Claim 3, wherein an acknowledgement transmission contains data appointing the time for an update command transmission and where the frequency for transmission is determined according to that time from the said pseudo random frequency hopping sequence.

6. A system as claimed in any preceding claim, in which several different times are appointed for multiple acknowledgements whereby each acknowledging radio transceiver chooses at random from those made available.

7. A system as claimed in any preceding claim, in which each radio transceiver includes partition size data in the acknowledgement transmission, the said partition size data being derived in dependence upon the number of radios, including itself, already synchronised.

8. A system as claimed in Claim 7, wherein the originator of the synchronisation transmission, upon the receipt of the acknowledgement transmissions, determines the common time on the basis of the partition size data contained in the acknowledgement transmission such that the common time is taken to be the time of a radio transceiver having the largest partition size.

9. A system as claimed in Claim 7, wherein the originator of the synchronisation transmission upon receipt of the acknowledgement transmission, determines the common time on the basis of the partition size data contained in the acknowledgment transmission, and wherein when no single largest partition obtains an average of the times is taken.

10. A system as claimed in claim 9, wherein the common time is taken as an average of the timings of the acknowledging radio

transceivers weighted linearly according to the partition sizes of the respective radio transceivers.

11. A system as claimed in any preceding claim, in which upon receipt of the update command transmission, an acknowledgement radio transceiver relays the update command transmission for receipt by other radio transceiver already synchronised to it.

12. A system as claimed in Claim 11, in which every radio transceiver receiving an initial update command transmission relays it at P random times.

13. A system as claimed in any preceding claim, in which prior to transmission of any update commands, a synchronisation transmission originator, determines an appointed time at which all updates should take place, which time is encoded into the update commands to ensure substantially simultaneous synchronisation operations.

14. A system as claimed in Claim 1, and substantially as hereinbefore described with reference to the accompanying drawings.

-38-

Patents Act 1977 Examiner's report to the Comptroller under Section 17 (The Search Report)	Application number 9113071.6
Relevant Technical fields (i) UK Cl (Edition K) H4L (LBSF) (ii) Int Cl (Edition 5) H04B, H04L, H04K	Search Examiner J BETTS
Databases (see over) (i) UK Patent Office (ii) On line databases: WPI	Date of Search 3 MARCH 1992

Documents considered relevant following a search in respect of claims 1-14

Category (see over)	Identity of document and relevant passages	Relevant to claim(s)
	NONE	

SF2(p)

Category	Identity of document and relevant passages	Relevant to claim(s)

Categories of documents

X: Document indicating lack of novelty or of inventive step.

Y: Document indicating lack of inventive step if combined with one or more other documents of the same category.

A: Document indicating technological background and/or state of the art.

P: Document published on or after the declared priority date but before the filing date of the present application.

E: Patent document published on or after, but with priority date earlier than, the filing date of the present application.

&: Member of the same patent family, corresponding document.

Databases: The UK Patent Office database comprises classified collections of GB, EP, WO and US patent specifications as outlined periodically in the Official Journal (Patents). The on-line databases considered for search are also listed periodically in the Official Journal (Patents).